



Fast Models

Version 11.26

Reference Guide

Non-Confidential

Copyright © 2017–2024 Arm Limited (or its affiliates).
All rights reserved.

Issue 00

100964_1126_00_en



Fast Models Reference Guide

This document is Non-Confidential.

Copyright © 2017–2024 Arm Limited (or its affiliates). All rights reserved.

This document is protected by copyright and other intellectual property rights.

Arm only permits use of this document if you have reviewed and accepted [Arm's Proprietary Notice](#) found at the end of this document.

This document (100964_1126_00_en) was issued on 2024-06-19. There might be a later issue at <http://developer.arm.com/documentation/100964>

The product version is 11.26.

See also: [Proprietary notice](#) | [Product and document information](#) | [Useful resources](#)

Start reading

If you prefer, you can skip to [the start of the content](#).

Intended audience

This document is written for software developers who are using the components and tools in the Fast Models portfolio to build custom platform models and integrate them with third-party tools and models.

Inclusive language commitment

Arm values inclusive communities. Arm recognizes that we and our industry have used language that can be offensive. Arm strives to lead the industry and create change.

This document includes language that can be offensive. We will replace this language in a future issue of this document.

To report offensive language in this document, email terms@arm.com.

Feedback

Arm welcomes feedback on this product and its documentation. To provide feedback on the product, create a ticket on <https://support.developer.arm.com>.

To provide feedback on the document, fill the following survey: <https://developer.arm.com/documentation-feedback-survey>.

Contents

1. About the models.....	21
1.1 Model capabilities.....	21
1.2 Running Arm Software Test Libraries (STL) on Fast Models.....	21
1.3 Quality level definitions.....	22
1.4 Fast Models accuracy.....	22
1.4.1 Timing accuracy of Fast Models.....	23
1.4.2 Bus traffic in Fast Models.....	23
1.4.3 Instruction prefetch in Fast Models.....	25
1.4.4 Out-of-order execution and write-buffers in Fast Models.....	25
1.4.5 Caches in Fast Models.....	26
1.4.6 Global exclusive monitor in Fast Models.....	27
1.5 Processor implementation.....	27
1.5.1 Mapping PMU events to MTI trace sources.....	27
1.5.2 Caches in PV models.....	29
1.5.3 GICv3 in PV models.....	31
1.5.4 GICv4 in PV models.....	31
1.5.5 CP14 Debug coprocessor.....	32
1.5.6 TLBs in PV models.....	32
1.5.7 Memory access in PV models.....	32
1.5.8 Timing in PV models.....	33
1.5.9 CADI interactions with processor behavior.....	34
1.6 Fast Models CADI implementation.....	35
1.7 CADI sync watchpoints.....	37
1.8 Non-CADI sync watchpoints.....	37
1.8.1 syncLevel definitions.....	38
1.8.2 Controlling and observing the syncLevel.....	39
1.9 SCADI.....	41
1.9.1 Intended uses of CADI and SCADI.....	41
1.9.2 Responsibilities of the SCADI caller.....	42
1.9.3 SCADI interface access.....	42
1.9.4 SCADI semantics.....	44
1.10 User mode networking.....	45

1.11 TAP/TUN networking.....	47
1.11.1 TAP/TUN networking limitations.....	47
1.11.2 Setting up a network connection for Microsoft Windows.....	47
1.11.3 Configuring the networking environment for Microsoft Windows.....	48
1.11.4 Commands for tap_setup_32.exe or tap_setup_64.exe for Microsoft Windows.....	48
1.11.5 Uninstalling networking for Microsoft Windows.....	49
1.11.6 Setting up a network connection for Red Hat Enterprise Linux.....	49
1.11.7 Setting up a network connection for Ubuntu Linux.....	51
1.11.8 Configuring the networking environment for Linux.....	53
1.11.9 Solutions to networking issues on Linux.....	54
1.11.10 Disabling and re-enabling networking for Linux.....	54
1.11.11 Uninstalling networking for Linux.....	55
1.12 Using parameters to set port values.....	55
1.13 PVBUS C++ transaction and Tx_Result classes.....	56
1.13.1 Class pv::TransactionGenerator.....	56
1.13.2 TransactionGenerator efficiency considerations.....	57
1.13.3 Enum pv::AccessWidth.....	57
1.13.4 Class pv::Transaction.....	57
1.13.5 Class pv::ReadTransaction.....	58
1.13.6 Class pv::WriteTransaction.....	58
1.14 Visualisation library.....	59
1.14.1 LISA visualisation models.....	60
1.14.2 Visualisation library C++ classes.....	60
2. Protocols.....	65
2.1 AMBA-PV protocols.....	65
2.1.1 AMBAPV protocol.....	65
2.1.2 AMBAPVACE protocol.....	68
2.1.3 AMBAPVSignal protocol.....	71
2.1.4 AMBAPVSignalState protocol.....	71
2.1.5 AMBAPVValue protocol.....	71
2.1.6 AMBAPVValue64 protocol.....	71
2.1.7 AMBAPVValueState protocol.....	72
2.1.8 AMBAPVValueState64 protocol.....	72
2.2 Clocking protocols.....	72
2.2.1 ClockRateControl protocol.....	72

2.2.2 TimerCallback protocol.....	73
2.2.3 TimerCallback64 protocol.....	73
2.2.4 TimerControl protocol.....	73
2.2.5 TimerControl64 protocol.....	74
2.3 Debug interface protocols.....	74
2.3.1 CADIDisassemblerProtocol protocol.....	75
2.3.2 CADIProtocol protocol.....	76
2.4 Peripheral protocols.....	78
2.4.1 AudioControl protocol.....	78
2.4.2 CharacterLCD protocol.....	79
2.4.3 FlashLoaderPort protocol.....	79
2.4.4 GUIPollCallback protocol.....	79
2.4.5 ICS307Configuration protocol.....	79
2.4.6 KeyboardStatus protocol.....	80
2.4.7 LCD protocol.....	80
2.4.8 LCDLayoutInfo protocol.....	81
2.4.9 MMC_Protocol protocol.....	81
2.4.10 MouseStatus protocol.....	82
2.4.11 PL080_DMACE_DmaPortProtocol protocol.....	82
2.4.12 PS2Data protocol.....	83
2.4.13 PVBUSSlaveControl protocol.....	83
2.4.14 PVDevice protocol.....	86
2.4.15 PVTransactionMaster protocol.....	86
2.4.16 SerialData protocol.....	87
2.4.17 SMMUv3AEMIdentifyProtocol protocol.....	88
2.4.18 TZFilterControl protocol.....	88
2.4.19 VirtualEthernet protocol.....	89
2.5 Power management protocols.....	89
2.5.1 PChannel protocol.....	89
2.6 Processor protocols.....	90
2.6.1 CoprocBusProtocol protocol.....	90
2.6.2 CounterInterface protocol.....	96
2.6.3 GICv3Comms protocol.....	96
2.6.4 InstructionCount protocol.....	97
2.6.5 v8EmbeddedCrossTrigger_controlprotocol protocol.....	98
2.7 Signaling protocols.....	98

2.7.1 Signal protocol.....	98
2.7.2 StateSignal protocol.....	98
2.7.3 Value protocol.....	99
2.7.4 Value_64 protocol.....	99
2.7.5 ValueState protocol.....	99
3. Fast Models components.....	100
3.1 Component differences.....	100
3.2 Bridge components.....	102
3.2.1 AMBAPV2PVBUS.....	103
3.2.2 AMBAPVACE2PVBUS.....	105
3.2.3 AMBAPVSignal2SGSignal.....	106
3.2.4 AMBAPVSignalState2SGStateSignal.....	106
3.2.5 AMBAPVValue2SGValue.....	107
3.2.6 AMBAPVValue2SGValue64.....	107
3.2.7 AMBAPVValue642SMMUv3AEMIdentify.....	108
3.2.8 AMBAPVValue642VECB.....	108
3.2.9 AMBAPVValueState2SGValueState.....	109
3.2.10 AMBAPVValueState2SGValueState64.....	110
3.2.11 BroadcastSignal2AMBAPVSignal.....	110
3.2.12 Clock2SystemC.....	111
3.2.13 ClockRateConversion.....	111
3.2.14 ClockSignal2SC_ClockSignal.....	113
3.2.15 CoprocBus2SystemC.....	114
3.2.16 CounterInterface2SystemC.....	114
3.2.17 InstructionCount2SystemC.....	115
3.2.18 LCD2SystemC.....	116
3.2.19 PChannel2SystemC.....	116
3.2.20 PVBUS2AMBAPV.....	117
3.2.21 PVBUS2AMBAPVACE.....	120
3.2.22 PVBUSBridge.....	123
3.2.23 SC_ClockSignal2ClockSignal.....	123
3.2.24 SGSignal2AMBAPVSignal.....	124
3.2.25 SGStateSignal2AMBAPVSignalState.....	124
3.2.26 SGValue2AMBAPVValue.....	125
3.2.27 SGValue2AMBAPVValue64.....	125

3.2.28 SGValueState2AMBAPVValueState.....	126
3.2.29 SGValueState2AMBAPVValueState64.....	127
3.2.30 SMMUv3AEMIdentify2AMBAPVValue64.....	127
3.2.31 SystemC2Clock.....	128
3.2.32 SystemC2Coprocbus.....	128
3.2.33 SystemC2CounterInterface.....	129
3.2.34 SystemC2InstructionCount.....	129
3.2.35 SystemC2LCD.....	130
3.2.36 SystemC2PChannel.....	131
3.2.37 SystemC2VirtualEthernet.....	131
3.2.38 SystemC2v7VGICConfig.....	132
3.2.39 VECB2AMBAPVValue64.....	132
3.2.40 VirtualEthernet2SystemC.....	133
3.2.41 v7VGICConfig2SystemC.....	133
3.3 Bus components.....	134
3.3.1 Labeller.....	135
3.3.2 LabellerForDMA330.....	136
3.3.3 LabellerForGPUProtMode.....	138
3.3.4 MSIRewriter.....	139
3.3.5 PASSwitch.....	142
3.3.6 PVBUS4KBT01KBSplitter.....	143
3.3.7 PVBUSCache.....	144
3.3.8 PVBUSDecoder.....	145
3.3.9 PVBUSExclusiveMonitor.....	145
3.3.10 PVBUSExclusiveSquasher.....	148
3.3.11 PVBUSLogger.....	148
3.3.12 PVBUSMapper.....	150
3.3.13 PVBUSMaster.....	150
3.3.14 PVBUSModifier.....	151
3.3.15 PVBUSRouter.....	152
3.3.16 PVBUSSlave.....	153
3.3.17 PVMemoryProtectionEngine.....	154
3.3.18 PVWriteBuffer.....	157
3.3.19 SimplePVBUSMaster.....	158
3.3.20 TZSwitch.....	159
3.4 Clocking components.....	161

3.4.1 ClockDivider.....	161
3.4.2 ClockGate_cpp.....	163
3.4.3 ClockSelector_cpp.....	164
3.4.4 ClockTimer.....	169
3.4.5 ClockTimer64.....	169
3.4.6 ClockTimerThread.....	170
3.4.7 ClockTimerThread64.....	171
3.4.8 MasterClock.....	171
3.4.9 PLLControl_cpp.....	172
3.4.10 ScalableClockControl_cpp.....	173
3.4.11 SwitchedClockControl_cpp.....	182
3.5 Core components.....	189
3.5.1 AEMv8RMPCT.....	190
3.5.2 AEMvACT.....	323
3.5.3 ARMAEMv8MCT.....	544
3.5.4 ARMCortexA5CT.....	778
3.5.5 ARMCortexA5MPx1CT.....	789
3.5.6 ARMCortexA7x1CT.....	802
3.5.7 ARMCortexA8CT.....	819
3.5.8 ARMCortexA9MPx1CT.....	832
3.5.9 ARMCortexA9UPCT.....	846
3.5.10 ARMCortexA15x1CT.....	858
3.5.11 ARMCortexA17x1CT.....	875
3.5.12 ARMCortexA32x1CT.....	892
3.5.13 ARMCortexA34x1CT.....	911
3.5.14 ARMCortexA35x1CT.....	930
3.5.15 ARMCortexA53x1CT.....	949
3.5.16 ARMCortexA55CT.....	970
3.5.17 ARMCortexA55CT_CortexA75CT.....	992
3.5.18 ARMCortexA55CT_CortexA76CT.....	1027
3.5.19 ARMCortexA55CT_CortexA78CT.....	1061
3.5.20 ARMCortexA57x1CT.....	1097
3.5.21 ARMCortexA65AECT.....	1116
3.5.22 ARMCortexA65AECT_CortexA76AECT.....	1141
3.5.23 ARMCortexA65CT.....	1178
3.5.24 ARMCortexA72x1CT.....	1201

3.5.25 ARMCortexA73x1CT.....	1220
3.5.26 ARMCortexA75CT.....	1239
3.5.27 ARMCortexA76AECT.....	1261
3.5.28 ARMCortexA76CT.....	1285
3.5.29 ARMCortexA77CT.....	1308
3.5.30 ARMCortexA78AECT.....	1330
3.5.31 ARMCortexA78CCT.....	1354
3.5.32 ARMCortexA78CT.....	1378
3.5.33 ARMCortexA510CT.....	1401
3.5.34 ARMCortexA510CT_CortexA710CT.....	1430
3.5.35 ARMCortexA510CT_CortexA710CT_CortexX2CT.....	1476
3.5.36 ARMCortexA510CT_CortexA710CT_CortexX3CT.....	1539
3.5.37 ARMCortexA510CT_CortexA715CT_CortexX3CT.....	1605
3.5.38 ARMCortexA520AECT.....	1672
3.5.39 ARMCortexA520CT.....	1703
3.5.40 ARMCortexA520CT_CortexA720CT.....	1735
3.5.41 ARMCortexA520CT_CortexA720CT_CortexX4CT.....	1783
3.5.42 ARMCortexA520CT_CortexA725CT.....	1851
3.5.43 ARMCortexA520CT_CortexA725CT_CortexX925CT.....	1899
3.5.44 ARMCortexA710CT.....	1964
3.5.45 ARMCortexA715CT.....	1992
3.5.46 ARMCortexA720AECT.....	2023
3.5.47 ARMCortexA720CT.....	2054
3.5.48 ARMCortexA725CT.....	2085
3.5.49 ARMCortexA725CT_CortexX925CT.....	2113
3.5.50 ARMCortexM0CT.....	2161
3.5.51 ARMCortexM0PlusCT.....	2166
3.5.52 ARMCortexM3CT.....	2173
3.5.53 ARMCortexM4CT.....	2179
3.5.54 ARMCortexM7CT.....	2186
3.5.55 ARMCortexM23CT.....	2197
3.5.56 ARMCortexM33CT.....	2212
3.5.57 ARMCortexM35PCT.....	2232
3.5.58 ARMCortexM52CT.....	2250
3.5.59 ARMCortexM55CT.....	2270
3.5.60 ARMCortexM85CT.....	2291

3.5.61 ARMCortexR4CT.....	2312
3.5.62 ARMCortexR5x1CT.....	2322
3.5.63 ARMCortexR7x1CT.....	2334
3.5.64 ARMCortexR8x1CT.....	2346
3.5.65 ARMCortexR52PlusCT.....	2360
3.5.66 ARMCortexR52x1CT.....	2376
3.5.67 ARMCortexR82AECT.....	2392
3.5.68 ARMCortexR82CT.....	2420
3.5.69 ARMCortexX1CCT.....	2447
3.5.70 ARMCortexX1CT.....	2470
3.5.71 ARMCortexX2CT.....	2493
3.5.72 ARMCortexX3CT.....	2520
3.5.73 ARMCortexX4CT.....	2551
3.5.74 ARMCortexX925CT.....	2580
3.5.75 ARMNeoverseE1CT.....	2608
3.5.76 ARMNeoverseN1CT.....	2631
3.5.77 ARMNeoverseN2CT.....	2654
3.5.78 ARMNeoverseN3CT.....	2681
3.5.79 ARMNeoverseV1CT.....	2708
3.5.80 ARMNeoverseV2CT.....	2731
3.5.81 ARMNeoverseV3AECT.....	2760
3.5.82 ARMNeoverseV3CT.....	2790
3.5.83 ARMSC000CT.....	2820
3.5.84 ARMSC300CT.....	2827
3.6 Media components.....	2833
3.6.1 D71.....	2833
3.6.2 DP500.....	2839
3.6.3 DP500x2.....	2842
3.6.4 DP550.....	2844
3.6.5 DP550x2.....	2847
3.6.6 DP650.....	2850
3.6.7 DP650x2.....	2853
3.6.8 Mali_C5x_streaming_sink.....	2856
3.6.9 Mali_C7x_streaming_sink.....	2858
3.6.10 Mali_C55.....	2859
3.6.11 Mali_C71.....	2863

3.6.12 Mali_C78.....	2865
3.6.13 Mali_Cxx_streaming_camera.....	2867
3.6.14 Mali_G51.....	2868
3.6.15 Mali_G71.....	2870
3.6.16 Mali_G72.....	2873
3.6.17 Mali_G76.....	2875
3.6.18 Mali_G78AE.....	2877
3.6.19 Mali_G710.....	2880
3.6.20 Mali_G715.....	2883
3.6.21 Mali_G720.....	2886
3.6.22 Mali_G725.....	2888
3.6.23 Mali_T624.....	2891
3.6.24 V61.....	2892
3.6.25 V550.....	2896
3.7 Peripheral components.....	2900
3.7.1 AudioOut_File.....	2900
3.7.2 AudioOut_SDL.....	2901
3.7.3 Base_PowerController.....	2901
3.7.4 Cluster_Temperature_Sensor.....	2904
3.7.5 CombinedMessagingUnit.....	2906
3.7.6 DebugAccessPort.....	2916
3.7.7 DebugROM.....	2917
3.7.8 DualClusterSystemConfigurationBlock.....	2929
3.7.9 DummyAPB.....	2933
3.7.10 ElfLoader.....	2936
3.7.11 FlashLoader.....	2938
3.7.12 GICv3CommsLogger.....	2939
3.7.13 GICv3CommsPVBUS.....	2940
3.7.14 GICv3ProtocolChecker.....	2941
3.7.15 GUIPoll.....	2942
3.7.16 HostBridge.....	2943
3.7.17 HostSerialInterface.....	2945
3.7.18 Infra6_SRAM_ECC_RAS.....	2945
3.7.19 IntelStrataFlashJ3.....	2946
3.7.20 Interrupt_Router.....	2950
3.7.21 LS64TestingFIFO.....	2952

3.7.22	MemoryMappedGenericTimer.....	2953
3.7.23	MemoryMappedGenericTimer_cpp.....	2956
3.7.24	MemoryMappedGenericWatchdog.....	2957
3.7.25	MemoryMappedGenericWatchdog_cpp.....	2959
3.7.26	NonVolatileCounter.....	2960
3.7.27	PCIeATC.....	2962
3.7.28	PLLClockControl.....	2963
3.7.29	PPUMultiThreadModeSwitch.....	2964
3.7.30	PS2Keyboard.....	2965
3.7.31	PS2Mouse.....	2966
3.7.32	PVBusGICv3Comms.....	2967
3.7.33	PVMetaDataController.....	2967
3.7.34	PchannelListener.....	2970
3.7.35	RAMDevice.....	2970
3.7.36	ROM.....	2972
3.7.37	RSS_Integ_Regs.....	2973
3.7.38	RandomNumberGenerator.....	2974
3.7.39	RealTimeLimiter.....	2975
3.7.40	RealtimeClockTimer.....	2977
3.7.41	RemapDecoder.....	2977
3.7.42	RootKeyStorage.....	2978
3.7.43	SI_System_Ctrl_Regs.....	2980
3.7.44	SMCF.....	2983
3.7.45	SMMUv3TestEngine.....	2989
3.7.46	STLBusGasket.....	2991
3.7.47	SerialCrossover.....	2992
3.7.48	TelnetTerminal.....	2993
3.7.49	UnusedPrimeCell.....	2997
3.7.50	VirtioBlockDevice.....	2998
3.7.51	VirtioBlockDeviceMMIO.....	3000
3.7.52	VirtioNetMMIO.....	3002
3.7.53	VirtioP9Device.....	3006
3.7.54	VirtioPCIBlockDevice.....	3008
3.7.55	VirtioRNG.....	3010
3.7.56	VirtualEthernetCrossover.....	3013
3.7.57	VirtualEthernetHub3.....	3014

3.7.58 VisEventRecorder.....	3014
3.7.59 Visualisation_sdl2_cpp.....	3017
3.7.60 WarningMemory.....	3026
3.7.61 v8EmbeddedCrossTrigger_Matrix.....	3028
3.8 Scheduler components.....	3029
3.8.1 AsyncSignal.....	3029
3.8.2 SchedulerInterface.....	3029
3.8.3 SchedulerThread.....	3030
3.8.4 SchedulerThreadEvent.....	3030
3.9 Signals components.....	3031
3.9.1 AndGate.....	3031
3.9.2 FrequencyProbe_cpp.....	3031
3.9.3 LabellerMasterIdExtendedIdUserFlag.....	3032
3.9.4 LabellerUserSignals.....	3035
3.9.5 OrGate.....	3036
3.9.6 SGSignalBuffer.....	3037
3.9.7 SignalDriver.....	3037
3.9.8 SignalInverter_cpp.....	3038
3.9.9 SignalLogger.....	3039
3.9.10 Value64Logger.....	3040
3.9.11 ValueLogger.....	3040
3.9.12 WideAndGate_cpp.....	3041
3.9.13 WideOrGate_cpp.....	3042
3.10 SystemIP components.....	3042
3.10.1 AHCI_SATA.....	3043
3.10.2 AddressTranslationUnit.....	3044
3.10.3 BP141_TZMA.....	3046
3.10.4 BP147_TZPC.....	3048
3.10.5 CCI400.....	3049
3.10.6 CCI500.....	3053
3.10.7 CCI550.....	3060
3.10.8 CCN502.....	3068
3.10.9 CCN504.....	3077
3.10.10 CCN508.....	3085
3.10.11 CCN512.....	3093
3.10.12 CI700.....	3101

3.10.13 CMN600.....	3137
3.10.14 CMN600AE.....	3148
3.10.15 CMN600CMLHub.....	3159
3.10.16 CMN650.....	3161
3.10.17 CMN650R2.....	3174
3.10.18 CMN700.....	3186
3.10.19 DMA350.....	3229
3.10.20 DMC500.....	3241
3.10.21 DMC520.....	3243
3.10.22 DMC620.....	3245
3.10.23 DMC_400.....	3247
3.10.24 EthosU55.....	3250
3.10.25 EthosU65.....	3252
3.10.26 EthosU85.....	3254
3.10.27 Firewall.....	3256
3.10.28 GIC500.....	3267
3.10.29 GIC500_ClusterPorts.....	3295
3.10.30 GIC500_Filter.....	3323
3.10.31 GIC600.....	3351
3.10.32 GIC600AE.....	3359
3.10.33 GIC600AE_Filter.....	3368
3.10.34 GIC600_Filter.....	3377
3.10.35 GIC625.....	3385
3.10.36 GIC625_Filter.....	3391
3.10.37 GIC700.....	3397
3.10.38 GIC700_Filter.....	3408
3.10.39 GIC720AE.....	3420
3.10.40 GIC720AE_Filter.....	3430
3.10.41 GIC_400.....	3441
3.10.42 GIC_IRI.....	3448
3.10.43 GIC_IRI_Filter.....	3485
3.10.44 ICS307.....	3521
3.10.45 IDAU.....	3524
3.10.46 IntegrityChecker.....	3756
3.10.47 KeyManagementUnit.....	3758
3.10.48 LifeCycleManager.....	3763

3.10.49 MMC.....	3768
3.10.50 MMU_400.....	3774
3.10.51 MMU_400_BASE.....	3801
3.10.52 MMU_500.....	3806
3.10.53 MMU_500_BASE.....	3837
3.10.54 MMU_600.....	3847
3.10.55 MMU_700.....	3871
3.10.56 MMU_S3.....	3899
3.10.57 MemoryMappedCounterModule.....	3929
3.10.58 MessageHandlingUnit.....	3933
3.10.59 MessageHandlingUnitV2.....	3938
3.10.60 MessageHandlingUnitV3.....	3940
3.10.61 NI700.....	3944
3.10.62 PL011_Uart.....	3949
3.10.63 PL022_SSP.....	3954
3.10.64 PL030_RTC.....	3956
3.10.65 PL031_RTC.....	3957
3.10.66 PL041_AACI.....	3959
3.10.67 PL050_KMI.....	3960
3.10.68 PL061_GPIO.....	3962
3.10.69 PL080_DMAC.....	3963
3.10.70 PL110_CLCD.....	3966
3.10.71 PL111_CLCD.....	3968
3.10.72 PL180_MCI.....	3969
3.10.73 PL192_VIC.....	3971
3.10.74 PL310_L2CC.....	3972
3.10.75 PL330_DMAC.....	3978
3.10.76 PL340_DMC.....	3988
3.10.77 PL350_SMC.....	3997
3.10.78 PL350_SMC_NAND_FLASH.....	4018
3.10.79 PL370_HDLCD.....	4021
3.10.80 PL390_GIC.....	4023
3.10.81 PPUMTWakerequest.....	4032
3.10.82 PPUv0.....	4033
3.10.83 PPUv1.....	4036
3.10.84 PPUv1_Cluster_Wakerequest_Logic.....	4043

3.10.85	RSS_CPU_Private_Region.....	4044
3.10.86	SMMUv3AEM.....	4046
3.10.87	SMSC_91C111.....	4099
3.10.88	SP804_Timer.....	4101
3.10.89	SP805_Watchdog.....	4103
3.10.90	SP810_SysCtrl.....	4104
3.10.91	SSU.....	4108
3.10.92	SecureAlarmManager.....	4109
3.10.93	SecureICache.....	4111
3.10.94	SystemFMU.....	4114
3.10.95	System_RAS_Agent.....	4115
3.10.96	TZC_400.....	4117
3.10.97	TZFilterUnit.....	4129
3.10.98	TZIC.....	4130
3.10.99	TrustedRAM.....	4131
3.10.100	VHT_VIOBridge.....	4132
3.10.101	VHT_VSIBridge.....	4133
3.10.102	VHT_VSocket.....	4134
3.10.103	v7_VGIC.....	4134
4.	Plug-ins for Fast Models.....	4138
4.1	Loading a plug-in.....	4138
4.1.1	--plugin command-line option.....	4138
4.1.2	scx::scx_load_plugin() method.....	4138
4.1.3	FM_TRACE_PLUGINS environment variable.....	4139
4.2	Customizing a plug-in.....	4140
4.3	ArchMsgTrace.....	4140
4.3.1	ArchMsgTrace - parameters.....	4142
4.4	ASTFplugin.....	4142
4.4.1	ASTFplugin - parameters.....	4143
4.4.2	ASTFplugin usage notes.....	4144
4.4.3	Additional ASTF support in Fast Models.....	4144
4.4.4	ASTF tools.....	4145
4.5	BranchPrediction.....	4146
4.5.1	BranchPrediction - parameters.....	4146
4.5.2	BranchPrediction output example.....	4147

4.5.3 Other ways to report branch mispredictions.....	4147
4.6 CADIIPCRemoteConnection.....	4148
4.6.1 CADIIPCRemoteConnection - parameters.....	4148
4.7 CDE.....	4149
4.7.1 CDETester.....	4149
4.7.2 CDETester - parameters.....	4151
4.7.3 CDEConstant.....	4152
4.7.4 CDEConstant - parameters.....	4152
4.7.5 Implementing a CDE plug-in.....	4152
4.7.6 CDE API.....	4155
4.8 Crypto.....	4162
4.8.1 Crypto - parameters.....	4164
4.9 GDBRemoteConnection.....	4165
4.9.1 GDBRemoteConnection limitations.....	4166
4.10 GenericCounter.....	4166
4.10.1 GenericCounter - parameters.....	4167
4.11 GenericTrace.....	4167
4.11.1 GenericTrace - parameters.....	4168
4.11.2 GenericTrace plug-in usage example.....	4170
4.11.3 Mapping between SYSREG_UPDATE trace sources and SPSR registers.....	4172
4.12 libete-plugin.....	4172
4.12.1 libete-plugin - parameters.....	4172
4.13 ListTraceSources.....	4174
4.13.1 ListTraceSources - parameters.....	4175
4.14 PipelineModel.....	4175
4.14.1 CortexA53PipelineModel - parameters.....	4177
4.14.2 InOrderPipelineModel - parameters.....	4177
4.14.3 PipelineModel example.....	4177
4.14.4 Naming the plug-in instance.....	4178
4.14.5 Example command lines.....	4179
4.14.6 PipelineModel output.....	4181
4.15 RunTimeParameterTest.....	4181
4.16 ScalableVectorExtension.....	4182
4.16.1 ScalableVectorExtension - parameters.....	4182
4.17 Sidechannel.....	4185
4.18 TarmacText.....	4186

4.18.1 TarmacText - parameters.....	4186
4.19 TarmacTrace.....	4187
4.19.1 TarmacTrace - parameters.....	4187
4.19.2 TarmacTrace file format.....	4189
4.19.3 Tarmac Trace output example.....	4191
4.19.4 Instruction trace.....	4193
4.19.5 Program flow trace.....	4194
4.19.6 Register trace.....	4195
4.19.7 Cache maintenance trace.....	4197
4.19.8 Cache content trace.....	4197
4.19.9 Translation table walk trace.....	4199
4.19.10 Granule protection table walk trace.....	4200
4.19.11 TLB trace.....	4201
4.19.12 Event trace.....	4202
4.19.13 Processor memory access trace.....	4204
4.19.14 Processor memory update trace.....	4205
4.19.15 Memory bus trace.....	4206
4.20 ToggleMTIPlugin.....	4209
4.20.1 ToggleMTIPlugin - parameters.....	4209
4.20.2 How to use ToggleMTIPlugin.....	4210
5. Fast Models examples.....	4211
5.1 CADI examples.....	4211
5.2 LISA examples.....	4212
5.3 Build and run an FVP example.....	4213
5.4 LISAPlus examples.....	4217
5.5 MTI examples.....	4217
5.6 SystemCExport examples.....	4218
5.7 Build and run an EVS platform example.....	4219
6. Base Platform.....	4221
6.1 Base Platform RevC.....	4222
6.1.1 Base Platform RevC block diagram.....	4223
6.1.2 BasePlatformPCIRevC component.....	4223
6.1.3 Base Platform RevC PCIeRootComplex.....	4226
6.1.4 Base Platform RevC SMMUv3AEM.....	4227
6.1.5 Base Platform RevC PCIe device assignment.....	4228

6.1.6 Base Platform RevC legacy PCI interrupts.....	4229
6.1.7 Base Platform RevC MSI-X.....	4229
6.2 Base Platform startup configuration for v9 cores.....	4229
6.3 Base Platform memory.....	4230
6.3.1 Base Platform memory map.....	4230
6.3.2 Base Platform secure memory.....	4233
6.3.3 BaseR Platform memory map.....	4233
6.3.4 Base Platform DRAM.....	4234
6.4 Base Platform interrupt assignments.....	4234
6.5 Base Platform clocks.....	4236
6.6 Base Platform parameters.....	4237
6.7 Base Platform components.....	4238
6.7.1 Base Platform Base_PowerController component.....	4238
6.7.2 Base Platform DebugAccessPort component.....	4243
6.7.3 Simulator visualization - parameters.....	4244
6.7.4 VE_SysRegs component.....	4245
6.8 Base Platform VE compatibility.....	4246
6.8.1 Base Platform GICv2.....	4246
6.8.2 Base Platform GICv3.....	4247
6.8.3 Base Platform system global counter.....	4247
6.8.4 Disable security in the Base Platform.....	4247
6.9 Base Platform unsupported VE features.....	4248
6.9.1 Base Platform memory aliasing at 0x08_00000000.....	4248
6.9.2 Boot ROM alias at 0x00_0800_0000 in the Base Platform.....	4248
6.9.3 Changed parameters in the Base Platform.....	4248
7. Microcontroller Prototyping System 2.....	4249
7.1 MPS2.....	4249
7.2 MPS2 platform types.....	4249
7.3 MPS2 memory maps.....	4250
7.3.1 MPS2 memory map for models without the Armv8-M additions.....	4250
7.3.2 MPS2 memory map for models with the Armv8-M additions.....	4251
7.4 MPS2 interrupt assignments.....	4257
7.5 MPS2 - differences between models and hardware.....	4257
8. Versatile Express Model.....	4260
8.1 VE.....	4260

8.2 VE memory map for Cortex-A series.....	4262
8.3 VE memory map for Cortex-R series.....	4264
8.4 VE interrupt assignments for Cortex-A series.....	4265
8.5 VE interrupt assignments for Cortex-R series.....	4266
8.6 VE parameters.....	4267
8.6.1 VE instantiation parameters.....	4267
8.6.2 VE secure memory parameters.....	4268
8.6.3 VE switch S6.....	4268
8.7 VE components.....	4269
8.7.1 VEVisualisation component.....	4269
8.8 VE_SysRegs component.....	4273
8.8.1 VE_SysRegs.....	4273
8.8.2 VE_SysRegs ports.....	4273
8.8.3 VE_SysRegs parameters.....	4274
8.8.4 VE_SysRegs registers.....	4274
8.8.5 VE_SysRegs - verification and testing.....	4275
8.9 Differences between the VE hardware and the system models.....	4275
8.9.1 Memory map.....	4275
8.9.2 Memory aliasing.....	4276
8.9.3 VE hardware features absent.....	4276
8.9.4 VE hardware features different.....	4276
Proprietary notice.....	4277
Product and document information.....	4279
Product status.....	4279
Revision history.....	4279
Conventions.....	4281
Useful resources.....	4283

1. About the models

Programmers View (PV) models of processors and devices work at a level where functional behavior is equivalent to what a programmer would see using the hardware.

They sacrifice timing accuracy to achieve fast simulation execution speeds. You can use the PV models for confirming software functionality, but you must not rely on the accuracy of cycle counts, low-level component interactions, or other hardware-specific behavior.

1.1 Model capabilities

Fast Models attempt to accurately model the hardware, but compromises exist between speed of execution, accuracy, and other criteria. A processor model might not match the hardware under certain conditions.

Fast Models can:

- Accurately model instructions.
- Correctly execute architecturally-correct code.
- Model some unpredictable behavior.

Fast Models cannot:

- Validate the hardware.
- Model all unpredictable behavior.
- Model cycle counting.
- Model timing-sensitive behavior.
- Model SMP instruction scheduling.
- Measure software performance.
- Model bus traffic.

Related information

- [Caches in Fast Models](#)

1.2 Running Arm Software Test Libraries (STL) on Fast Models

Fast Models does not support running the entire STL.

The reasons are:

- Some micro-architectural features implemented in the RTL are not implemented in the Fast Model and therefore the model's behavior might deviate from the RTL.
- Some peripheral support must be included to enable parts of the STL. For example SBIST controller support is available in the RTL but not in the Fast Model.

Fast Models can be used for integration of the STL into your software stack, but running the STL might not perform accurate measurement and validation.

Related information

- [Arm Software Test Libraries](#)

1.3 Quality level definitions

The documentation for each model of Arm® IP includes one or more quality level statements to indicate how complete the model's implementation is for each supported revision of the IP.

Table 1-1: Quality level definitions

Quality level	Definition
Alpha support	The model implementation is at an early stage and is likely to change in future releases. There might be significant defects or limitations in the model.
Preliminary support	The model implementation is complete or almost complete. Testing is nearly to the level of a fully-supported model. However, changes might still occur and there are likely to be known defects present.
Full support	Also called Release quality. Support is complete and the model has been tested as fully as possible. The modeled IP has reached its LAC milestone.

1.4 Fast Models accuracy

Fast Models aim to be accurate from the point of view of the program running on the processors. This section describes areas where Fast Models differ from hardware.

Software is able to detect differences between hardware and the model, but these differences generally depend on behavior that is not precisely specified. For example, it is possible to detect differences in the exact timings of instructions and bus transactions, effects of speculative prefetch and cache victim selection.

Certain classes of behavior are specified as unpredictable and these cases are detectable by software. A program that relies on such behavior, even unintentionally, is not guaranteed to work reliably on any device, or on a Fast Model. Programs that exploit this behavior might execute differently between the hardware and the model.

Fast Models do not attempt to accurately model bus transactions. PVBUS provides the infrastructure to ensure that the program gets the correct results.

1.4.1 Timing accuracy of Fast Models

Fast Models do not model instruction timing accurately. The simulation as a whole has a very accurate concept of timing, but the Code Translation (CT) processors do not claim to dispatch instructions with device-like timing.

In general, a processor issues a set of instructions, called a quantum, at the same point in simulation time, and then waits for some amount of time before executing the next quantum. The timing is arranged so that on average the processor executes one instruction per clock cycle.



If timing annotation is enabled, the cycle count and instruction count can differ. For more details, see [Timing Annotation](#) in the Fast Models User Guide.

The consequences of this are:

- The perceived performance of software running on the model differs from real-world software. In particular, memory accesses and arithmetic operations all take a significant amount of time.
 - A program might be able to detect the quantized execution behavior of a processor, for example by polling a high-resolution timer.
 - All instructions in a quantum are effectively atomic.
-



This might mask some race-condition bugs in software.

1.4.2 Bus traffic in Fast Models

PVBus can simulate the behavior of individual bus transactions passing through a hierarchy of bus fabric, but it uses the following techniques to optimize this process:

- PVBus generally decodes the path between a bus master and bus slave the first time a transaction is issued. All subsequent transactions to the same address are automatically sent to the same slave, without passing through the intervening fabric.
- For accesses to normal memory, the master can cache a pointer to the host storage that holds the data contents of the memory. The master can read and write directly to this memory without generating bus transactions.
- For instruction-fetch, and for operations such as repeated DMA from framebuffer memory, PVBus provides an optimization called snooping, which informs the master if anyone else could have modified the contents of memory. If no changes have occurred, the master can avoid the need to re-read memory contents.

If a piece of bus fabric wants to intercept and log all bus transactions, it can defeat these optimizations by claiming to be a slave device. It can then log all transactions and can reissue identical transactions on its own master port. However, doing this slows all bus transactions and significantly impacts simulation performance.



If direct accesses to memory by the CT engine are intercepted by the fabric, the processor is forced to single step. Execution is much slower than normal operation with translated code.

The bus traffic generated by a processor is not representative of real traffic:

Timing differences

Reordering and buffering of memory accesses, out-of-order execution, speculative prefetch, and drain-buffers can cause timing differences. They are not modeled, since they are not visible to the programmer except in situations where a cluster program contains race conditions that violate serial-consistency expectations.

Bus contention

Fast Models do not model the time taken for a bus transaction, so they cannot model the effects of multiple transactions contending for bus availability.

Size of access

Fast Models do not attempt to generate the same types of burst transaction from the processor for accesses to multiple consecutive locations. PVBUS only supports burst transactions of type INCR.

Instruction fetch

The behavior of the instruction prefetch unit of a processor is not modeled to match the hardware implementation.

Behavioral differences

In some software, the trace of instruction execution depends on timing effects. For example, if a loop polls a device waiting for a 10ms time-out, the number of iterations of the polling loop depends on the rate of instruction execution.

Other differences

- PVBUS does not support any type of write strobes.
- PVBUS does not support Quality of Service (QoS) or region values.
- Transactions cannot cross a 4KB boundary.
- Barriers and DVM messages are not transmitted on the PVBUS.

Related information

[Instruction prefetch in Fast Models](#) on page 24

1.4.3 Instruction prefetch in Fast Models

The CT engine in the processor models relies on Fast Models PVBUS optimizations. It only performs code-translation if it has been able to prefetch and snoop the underlying memory. It then need not issue bus transactions until the snoop handling detects an external modification to memory.

If the CT engine cannot get prefetch access to memory, it drops to single-stepping. This single-stepping is very slow (~1000x slower than translated code execution).

Real processors attempt to prefetch instructions ahead of execution and predict branch destinations to keep the prefetch queue full. The instruction prefetch behavior of a processor can be observed by a program that writes into its own prefetch queue (without using explicit barriers). The architecture does not define the results.

The CT engine processes code in blocks. The effect is as if the processor filled its prefetch queue with a block of instructions, then executed the block to completion. As a result, this virtual prefetch queue is sometimes larger and sometimes smaller than the corresponding hardware. In the current implementation, the virtual prefetch queue can follow small forward branches.

With an L1 instruction cache turned on, the instruction blocksize is limited to a single cache-line. The processor ensures that a line is present in the cache at the point where it starts executing instructions from that line.

In real hardware, the effects of the instruction prefetch queue are to cause additional fetch transactions. Some of these are redundant because of incorrect branch prediction. This causes extra cache and bus pressure.

1.4.4 Out-of-order execution and write-buffers in Fast Models

Hardware memory is Weakly Ordered, but Fast Models memory is Strongly Ordered.

The CT implementation executes instructions sequentially. One instruction is retired before the next starts to execute. In a real processor, multiple memory accesses can be outstanding, and can complete in a different order from their program order. Writes can also be delayed in a write-buffer.

The programmer-visible effects of these behaviors are defined in the architecture as the Weakly Ordered memory model, which the programmer must be aware of when writing lock-free cluster code.

Within Fast Models, memory accesses happen in program order, effectively as if all memory is Strongly Ordered.

1.4.5 Caches in Fast Models

Fast Models with cache-state modeling enabled can replicate some types of failure-case, but not all types.

The effects of caches are programmer-visible because they can cause a single memory location to exist as multiple inconsistent copies. If caches are not correctly maintained, reads can observe stale copies of locations, and flushes/cleans can cause writes to be lost.

There are three ways in which incorrect cache maintenance can be programmer-visible:

From the D-side interface of a single processor

The only way of detecting the presence of caches is to create aliases in the memory map, so that the same range of physical addresses can be observed as both cached and non-cached memory.

From the D-side of a single processor to its I-side

Stale instruction data can be fetched when new instructions have been written by the D-side. This can either be because of deliberate self-modifying code, or as a consequence of incorrect OS demand paging.

Between one processor and another device

For example, another processor in a non-coherent MP system, or an external DMA device.

Fast Models with cache-state modeling enabled can replicate all of these failure cases. However, they do not attempt to reproduce the following effects of caches:

- Changes to timing behavior of a program because of cache hits/misses (because the timing of memory accesses is not modeled).
- Ordering of line-fill and eviction operations.
- Cache usage statistics (because the models do not generate accurate bus traffic).
- Device-accurate allocation of cache victim lines (which is not possible without accurate bus traffic modeling).
- Write-streaming behavior where a cache spots patterns of writes to consecutive addresses and automatically turns off the write-allocation policy.

The Cortex-A9 and Cortex-A5 models do not model device-accurate MESI behavior. The Cortex-A15 and Cortex-A7 models do simulate hardware MOESI state handling, and can handle cache-to-cache snoops. In addition, they model the AMBA 4 ACE cache-coherency protocols over their PVBUS ports, so can be connected to a model of an ACE Coherent Interconnect (such as the CCI-400 model) to simulate coherent sharing of cache contents between processors.

It is not possible to insert devices between the processor and its L1 caches. In particular, you cannot model L1 traffic, although you can tell the model not to model the state of L1 caches.

1.4.6 Global exclusive monitor in Fast Models

A monitor for cacheable nonshared and shared memory is always implemented, but the implementation differs depending on whether `cache-state-modelled` is `true` or `false`.

- When `cache-state-modelled` is `true`, caches are modeled, so exclusiveness is handled by the cache coherency protocol. Each line in a coherent cache records whether it is exclusive. When another core writes to an exclusive line, it is invalidated in other caches.
- When `cache-state-modelled` is `false`, the cache state is not modeled, so there are no cache lines or coherency. To provide the same functionality from a software perspective, an exclusive monitor is instantiated internally to maintain coherency.

The `RAMDevice` component and `PVBus` to `AMBAPV` bridges do not contain global monitors. As a result, exclusive stores that reach a `RAMDevice` fail unless they go through an exclusive monitor. Some platforms might need a global monitor outside of the cache coherency domains. These platforms must include a system-level monitor in the same place in the bus hierarchy as in the hardware. See the `FVP_VE` and `FVP_Base` example platforms for examples of how to use the `PVBusExclusiveMonitor` component.

1.5 Processor implementation

This section outlines the differences between the code translation models and the hardware.

1.5.1 Mapping PMU events to MTI trace sources

In Fast Models, many PMU events are modeled and are exposed through the standard PMU interface, although in many cases they are implemented using MTI trace sources.



Note

- Counters that are modeled can generate PMU overflow interrupts, which work the same as in hardware.
- Counters that are not modeled have the value zero and do not increment.

The following tables show which PMUv1 and PMUv2 events are modeled, and which MTI trace sources are used to model them. The associated event numbers are as listed in the Arm® Architecture Reference Manual Arm®v7-A and Arm®v7-R edition, section C12.8.2 Common event numbers.

Table 1-2: PMUv1 event mappings

PMUv1 event	MTI trace sources	Notes
Software increment (0x00)	-	Modeled, but not using MTI.
Level 1 instruction cache refill (0x01)	ALLOC_LINEFILL	-
Level 1 instruction TLB refill (0x02)	MMU_TLB_FILL	-
Level 1 data cache refill (0x03)	ALLOC_LINEFILL	-

PMUv1 event	MTI trace sources	Notes
Level 1 data cache access (0x04)	CACHE_READ_HIT, CACHE_READ_MISS, CACHE_WRITE_HIT, CACHE_WRITE_MISS	-
Level 1 data TLB refill (0x05)	MMU_TLB_FILL	-
Load (0x06)	CORE_LOADS, NV2_CORE_LOADS, MEMTAG_LOAD_INST, SVE_LD_RETIRED	-
Store (0x07)	CORE_STORES, NV2_CORE_STORES, MEMTAG_STORE_INST, SVE_ST_RETIRED	-
Instruction architecturally executed (0x08)	-	Modeled, but not using MTI.
Exception taken (0x09)	EXCEPTION	-
Exception return (0x0A)	EXCEPTION_RETURN_PREBRANCH	-
Write to CONTEXTIDR (0x0B)	CONTEXTIDR	-
Software change of the PC (0x0C)	BRA_DIR, BRA_INDIR.	-
Immediate branch (0x0D)	BRA_DIR, BRA_DIR_FAIL.	-
Procedure return (0x0E)	BRA_RET	-
Unaligned load or store (0x0F)	UNALIGNED_LDST_RETIRED, SVE_UNALIGNED_LDST_RETIRED	-
Mispredicted or not predicted branch speculatively executed (0x10)	BRANCH_MISPREDICT	Only if the BranchPrediction plug-in is loaded.
Cycle count (0x11)	-	Modeled, but not using MTI. Note: This value is an approximation.
Predictable branch speculatively executed (0x12)	-	Not modeled.

Table 1-3: PMUv2 event mappings

PMUv2 event	MTI source	Notes
Data memory access (0x13)	-	Not modeled.
Level 1 instruction cache access (0x14)	CACHE_READ_HIT, CACHE_READ_MISS	-
Level 1 data cache write-back (0x15)	-	Not modeled.
Level 2 data cache access (0x16)	CACHE_READ_HIT, CACHE_READ_MISS, CACHE_WRITE_HIT, CACHE_WRITE_MISS	-
Level 2 data cache refill (0x17)	ALLOC_LINEFILL	-
Level 2 data cache write-back (0x18)	-	Not modeled.
Bus access (0x19)	CORE_LOADS, CORE_STORES, MEMTAG_LOADS, MEMTAG_STORES	-
Memory error (0x1A)	-	Not modeled.
Instruction speculatively executed (0x1B)	-	Not modeled.
Write to TTBR (0x1C)	TTBR_WRITE	-

PMUv2 event	MTI source	Notes
Bus cycle (0x1D)	INST	This value is an approximation.

Related information

[ARM Architecture Reference Manual ARMv7-A and ARMv7-R edition](#)

1.5.2 Caches in PV models

Some processor models have PV-accurate caches, but others do not model Level 1 or Level 2 caches.

Cores that have PV-accurate cache implementation provide a functionally-accurate model. For more information, see the processor component descriptions.

Other PV models do not model Level 1 or Level 2 caches. The system coprocessor registers related to cache operations permit cache-aware software to work, but in most cases they only check register access permissions.

The registers affected on all code translation processor models are:

- Invalidate and/or Clean Entire ICache/DCache.
- Invalidate and/or Clean ICache/DCache by MVA.
- Invalidate and/or Clean ICache/DCache by Index.
- Invalidate and/or Clean Both Caches.
- Cache Dirty Status.
- Data Write Barrier.
- Data Memory Barrier.
- Prefetch ICache Line.
- ICache/DCache lockdown.
- ICache/DCache master valid.

1.5.2.1 Functional caches in Fast Models

Fast Models implement two types of cache model: register model and functional model.

A register model provides all the cache control registers so that cache operations succeed, but does not model the state of the cache. All accesses go to memory.

A functional model tracks cache state and its contents at each level of the memory hierarchy. Incorrect cache management might return incorrect data, as it would on real hardware.

Fast Models provide:

- System IPs that support caches.
- Register models of caches on all processors that support caches and also the PL310 cache controller (PL310_L2CC).
- Functional models of caches integrated into Cortex cores.

For a core with no L2 cache, the configuration parameters are:

icache-state-modelled

Set whether the I-cache has stateful implementation.

dcache-state-modelled

Set whether the D-cache has stateful implementation.

For Arm®v7-A cores with an L2 cache, the configuration parameters are:

l1_icache-state-modelled

Set whether L1 I-cache has stateful implementation.

l1_dcache-state-modelled

Set whether L1 D-cache has stateful implementation.

l2_cache-state-modelled

Enable unified Level 2 cache state model.

For Arm®v8-A cores with an L2 cache, the configuration parameters are:

icache-state_modelled

Set whether L1 I-cache has stateful implementation. Instructions at L2 or L3 are not cached in Fast Models.

dcache-state_modelled

Set whether D-cache has stateful implementation at L1, L2, and L3.

1.5.2.2 Performance impact of functional caches in Fast Models

Enabling functional cache modeling is likely to reduce performance.

Enable the L1 and L2 functional caches together. For consistent system operation, Arm recommends that you either disable functional behavior completely or enable it for both I and D L1 caches and the L2 cache.

Cache enablement must be system wide. If you enable cache state modeling in any component then you must enable it in all components in the system, including all cores (L1 and L2) and any external cache controller (such as the PL310_L2CC) and any interconnect that has caches.

If platform memory is being modeled outside of the Fast Models environment, for example in a SystemC environment, use of functional cache modeling might improve performance if there is no other fast route to memory.

1.5.3 GICv3 in PV models

The PV models implement the Generic Interrupt Controller architecture version 3 (GICv3).

The GICv3 architecture defines two parts:

- The core interface (integrated into the core)
- The Interrupt Redistribution Infrastructure (IRI)

You can configure all Arm®v8-A cores to include a GICv3 interface. You can integrate a separate GIC_IRI component into your platforms. Communication between the core and the IRI is over an architected packet interface. An internal communication protocol represents the packets that pass over this interface.

You can configure the GICv3 models in some platforms to act as though they were GICv2 or GICv2-M models. Even in this mode, you need a GIC_IRI component and a supported core. Configure them to comply with the same standard.

Models have the following limitations:

- Support for the GITS_CTLR.Quiescent bit is not complete.
- Support for ITS save/restore is not complete. Configuration stays within the model and it does not use allocated memory.
- GICD_CTLR.RWP does not perform adequately. This difference is only an issue if you use the distributor in systems with delaying interface between the distributor and the cores. Do not use this version of the model for simulation of the GIC in a setup where interfaces are not instantly reactive.

Set the environment variable `FASTSIM_GIC_MEMORY_MAP` to 1 to print to `stderr` the memory map of certain models that are included in the platform being run. This functionality is available for all GICv3 and later models.

1.5.4 GICv4 in PV models

GICv4 is an extension of the GICv3 architecture. It allows the direct injection of LPIs into a virtualized system through the `virtual-lpi-support` parameter of the `GIC_IRI` or `GIC_IRI_Filter` component.

In addition to requiring the presence of an ITS that is configured as shown in [GIC_IRI](#), GICv4 requires you to enable the virtual LPIs feature and to configure a virtual PE table using the parameters shown in this example:

```
"virtual-lpi-support"=true,  
"GITS_BASER4-type"=2 //Type 2 is Virtual PEs.  
                      //Such a table is needed for GICv4 functionality.
```

1.5.5 CP14 Debug coprocessor

Some models fully implement the CP14 Debug coprocessor registers. Other models only implement the DSCR register. This register reads as 0 and ignores writes.

External debugging must be used to debug systems containing PV models.

1.5.6 TLBs in PV models

The PV models implement Translation Lookaside Buffers (TLBs) and model most aspects of TLB behavior.



If the `device-accurate-tlb` parameter is set to `false`, the simulation uses a different number of TLBs if this improves simulation performance. The simulation is architecturally accurate, but not device accurate. Architectural accuracy is almost always sufficient. Set `device-accurate-tlb` to `true` if you require device accuracy.

These TLB registers do not have working implementations:

- Primary memory remap register.
- Normal memory remap register.

In addition, the simulation does not distinguish peripheral accesses from data accesses, so it ignores configuration of the peripheral port memory remap register.



The models do not implement device-accurate MicroTLBs, or system coprocessor registers related to MicroTLB state.

1.5.7 Memory access in PV models

PV models use a PVBUSMaster subcomponent to communicate with slaves in a System Canvas-generated system. This provides efficient access to memory-like slaves and relatively efficient access to device-like slaves.

Memory access in PV models differs from real hardware to enable fast modeling of the processor:

- All memory accesses are performed in programmer view order.
- Unaligned accesses, where permitted, are always performed as byte transfers.

In addition, some PV models do not use all the transaction states available in a PVBUS transaction. The Privileged and Instruction flags are set correctly for Arm®v7 processors but might not be set correctly in earlier architectures. However all memory accesses are atomic so `swp` instructions behave as expected.

1.5.7.1 I-side access in PV models

PV models cache translations of instructions fetched from memory-like slaves. The models might not perform further access to those slaves for significant periods. A slave can force the model to reread the memory by declaring that the memory has changed.

PV models do not model a prefetch queue but the code translation mechanism effectively acts as a prefetch queue of variable depth. Arm recommends that you follow the standards in the [Arm® architecture specifications](#) for dealing with prefetch issues, such as self modifying code, and use appropriate cache flushing and synchronization barriers.

Translation of instructions only occurs for memory-like slaves, which are those declared by devices as having type `PV::MEMORY`. Instructions fetched from device-like slaves are repeatedly fetched, decoded, and executed, significantly slowing down model performance.

1.5.7.2 D-side access in PV models

PV models cache references to the underlying memory of memory-like slaves, and might not perform further accesses to those slaves over the bus for significant periods.

Slaves declared as type `PV::MEMORY` provide the fastest possible memory access for PV processors.

Slaves declared as type `PV::DEVICE` are normally used for peripheral access.

1.5.8 Timing in PV models

Programmers View (PV) models are loosely timed.

- Caches and write buffers are not modeled, so all memory access timing is effectively zero wait state.
- All instructions execute, in aggregate, in one cycle of the component master clock input.
- Interrupts are not taken at every instruction boundary.
- Some sequences of instructions are executed atomically, ahead of the master clock of a component, so that system time does not advance during execution. This difference in behavior can affect sequential access of device registers, where devices are expecting time to move on between accesses.
- DMA to and from Tightly Coupled Memory (TCM) is atomic.

1.5.9 CADI interactions with processor behavior

Architecturally, M series processors have different reset behavior to that of A and R series processors.

For both hardware and model processors, a reset consists of asserting, and then deasserting the reset pin. However:

M series

The architecture documentation specifies that on asserting the reset pin, the processor stops executing instructions. On deasserting the reset pin, the VTOR is given its reset value, SP_main is read from address VTOR+0, and the PC is read from address VTOR+4. See the Reset behavior section and the Vector Table Offset Register, VTOR section in the [ARMv7-M Architecture Reference Manual](#). Also, the processor begins fetching and executing instructions.

A and R series

Asserting the reset pin causes the processor to stop executing instructions and the PC to be given its reset value, which depends on VINITHI, and the SP is UNKNOWN. See the Reset section in the [ARM Architecture Reference Manual ARMv7-A and ARMv7-R edition](#). On deasserting the reset pin, the processor begins fetching and executing instructions.

The behavior of reset on M series processors interacts differently with CADI debugging functionality than it does for A and R series processors.

On A and R series processors, after the reset pin of the processor is asserted, a new application can be loaded using the `CADIExecLoadApplication()` call in CADI. This loads an application into memory, and sets the PC if a start address is specified in the application. When reset is deasserted, the processor begins fetching and executing from the start address as expected. Similarly, if a debugger asserts reset on the processor, it can modify memory with a sequence of `CADIMemWrite()` calls, update the PC with a `CADIRegWrite()` call, and when reset is deasserted the processor begins to fetch and execute from the PC.

On M series processors, these techniques do not work because after reset is deasserted the processor updates SP_main and the PC, overwriting the settings made by the CADI calls.

For these techniques to be possible, the M series processor models differ slightly from the architectural reset behavior. When reset is asserted, the M series makes note of whether the PC or SP is modified before reset is next deasserted. If there are any CADI writes to the PC or SP registers, either directly through `CADIRegWrite` or indirectly through `CADIExecLoadApplication()`, this is tracked. When reset is deasserted, if the PC has been modified using CADI, the PC retains the value written to it, otherwise it reads it from address VTOR+4. Similarly, if the SP has been modified using CADI, SP_main retains the value written to it, otherwise it reads it from address VTOR+0.

1.6 Fast Models CADI implementation

Fast Models implement a subset of CADI functionality.

The following CADI methods are not supported by Fast Models:

CADI class:

Register API

`CADIGetCommittedPCs()`

Memory API

`CADIMemGetOverlays()`

Virtual Memory API

`PhysicalToVirtual()`

Cache API

- `CADIGetCacheInfo()`
- `CADICacheRead()`
- `CADICacheWrite()`

Execution API

- `CADIExecLoadApplication()`. Implemented by CPU components only
- `CADIExecUnloadApplication()`
- `CADIExecGetLoadedApplications()`. Implemented by CPU components only. It only returns the most recently loaded application if more than one have been loaded.
- `CADIExecAssertException()`
- `CADIExecGetPipeStages()`
- `CADIExecGetPipeStageFields()`
- `CADIGetCycleCount()`

CADICallbackObj class:

- `appliOpen()`
- `appliClose()`
- `cycleTick()`
- `killInterface()`

CADIDisassembler class:

- `GetSourceReferenceForAddress()`
- `GetAddressForSourceReference()`
- `GetInstructionType()`

CADIDisassemblerCB class:

- `ReceiveSourceReference()`

The following tables describe the implementation of other CADI functionality by processor models:

Table 1-4: CADI broker implementation

Feature	Implemented	Remarks
Factories	Yes	Model DLL provides a single factory.
Instances	Yes	Factory can only instantiate one model at a time.

Table 1-5: CADI target implementation

Feature	Implemented	Remarks
# breakpoints	Yes	No intrinsic limit.
Breakpoint disabling	Yes	You can disable global breakpoints, but not user breakpoints set through <code>CADIBptSet()</code> . Debuggers usually disable breakpoints with <code>CADIBptClear()</code> and re-enable them with <code>CADIBptSet()</code> , so this does not affect debugger GUIs, but it does affect the CADI C++ interface. Global breakpoints are built-in and permanent, so <code>CADIBptSet()/Clear()</code> cannot set or delete them. With <code>CADIBptConfigure()</code> , you can only enable/disable these global breakpoints to stop on certain processor events.
Breakpoint ignore count	No	-
Breakpoint formal conditions	No	-
Breakpoint free-form conditions	No	-
Breakpoint setting	Yes, but not permitted while target is running.	-
Breakpoint types	Yes: <code>CADI_BPT_MEMORY</code> , <code>CADI_BPT_PROGRAM</code> , <code>CADI_BPT_PROGRAM_RANGE</code>	All registers in the Vector register group have disabled register breakpoints. You cannot create new register breakpoints.
Breakpoint triggers	Yes: <code>CADI_BPT_TRIGGER_ON_MODIFY</code> , <code>CADI_BPT_TRIGGER_ON_READ</code> , <code>CADI_BPT_TRIGGER_ON_WRITE</code>	<code>CADI_BPT_TRIGGER_ON_WRITE</code> approximates <code>CADI_BPT_TRIGGER_ON_MODIFY</code> . If either is set, the simulation stops when a value is written, even when it does not change.
<code>CADIExecSingleStep()</code>	Yes	Fast Models ignores the <code>stepCycle</code> and <code>stepOver</code> arguments.
Intrusive debug	Yes	Stop/start can affect processor scheduling: single stepping and multiple stepping both change scheduling, and so are intrusive.
Memory read/write	Yes, but not permitted while target is running.	-
Reset	Yes	One reset level is supported. When calling <code>CADIExecReset()</code> , Arm recommends the simulation should use a <code>syncLevel</code> of 1 or higher for best results.
Runnable	Yes	-
Stop simulation	Yes	-

1.7 CADI sync watchpoints

The CADI/Synchronous CADI (SCADI) interfaces of processor components support watchpoints. Memory components like RAMDevice do not.

When the simulator hits a watchpoint, the CADI/SCADI interface emits a sequence, typically only one, of `modeChange (CADI_EXECMODE_Bpt, bptNumber)` callbacks, where `bptNumber` is the breakpoint ID of the watchpoint. After this sequence it sends a `modeChange (CADI_EXECMODE_stop)` callback. Only after the debugger receives this `stop` callback might it inspect the state of the model.

You can read additional information about the last of the hit watchpoints from these CADI registers declared by the processor in register group `simulation`:

memoryBptPC

The PC of the instruction that caused the watchpoint to hit.

memoryBptAccessVA

Virtual address of the access or sub-access that caused the watchpoint to hit.

memoryBptAccessSize

Size in bytes of the access or sub-access that caused the watchpoint to hit.

memoryBptAccessRW

Type of access or sub-access that caused the watchpoint to hit: 1 = read, 2 = write, 3 = both, 0 = no access.

These registers are not memory (or CPnn-) mapped anywhere and are not accessible to target programs. These registers are read-only. `memoryBptPC` and `memoryBptAccessVA` are 32 or 64 bit depending on the architecture. `memoryBptAccessSize` and `memoryBptAccessRW` are 32 bit.

When multiple accesses have hit watchpoints at the same time, for example during the same instruction, the information contained in these registers is valid and consistent only for one of these accesses.

Whenever at least one CADI watchpoint is set on a processor, the `syncLevel` on that processor is at least 2.

1.8 Non-CADI sync watchpoints

This section describes `syncLevel` enum values, semantics, and behaviors, with `syncLevel` values listed from fast to slow simulation, from fewer to more requirements.

1.8.1 syncLevel definitions

Definitions for the possible syncLevel values.

syncLevel 0: OFF

- The simulator runs as fast as possible. It does not permit inspection of the processor registers while the simulation is running, and does not stop synchronously when requested to do so.
- After enabling or disabling a trace source, it is undefined how many instructions will be executed before the change takes effect.
- Quantum end detection guarantees that a quantum is not overshoot indefinitely. Quantum end detection applies to (but is not limited to) backward branches, indirect jumps, exceptions, and atomic operation retries. In addition to temporal quantum end detection, some events may end a quantum, like executing a barrier, entering a low power state, or accessing a peripheral. Target software and simulation controllers must not rely on a specific scheduling pattern based on these quantum end check points.
- Use cases: normal fast simulation and normal CADI debugging while no CADI watchpoint is set.



When using CADI reset on a model, for best results, Arm recommends using a syncLevel of 1 or higher.

syncLevel 1: SYNC_STATE

- The simulation runs slightly slower than syncLevel 0. SCADI can read the up-to-date values of the processor registers, including PC and instruction count. You cannot stop the simulation synchronously.
- After enabling or disabling a trace source, it is undefined how many instructions will be executed before the change takes effect.
- Quantum end detection is as for syncLevel 0.
- Use cases: external breakpoints that block the simulation, inspect state of processor/memory from within a peripheral or memory access.

syncLevel 2: POST_INSN_IO

- As for syncLevel 1, except that you can stop the simulation synchronously from within all LD/ST and similar instructions. The simulation stops immediately after the current LD/ST instruction has been completely executed (post instruction).
- After enabling or disabling a trace source, it is likely, but not guaranteed, that the change will be visible sooner than with syncLevel 0 or 1.
- Quantum end detection is as for syncLevel 1, plus it includes the end of LD/ST instructions.
- Use cases: CADI watchpoints, external breakpoints, stopping from within LD/ST-related MTI callbacks.

syncLevel 3: POST_INSN_ALL

- As for syncLevel 2, except that you can stop the simulation synchronously from within any instruction. The simulation stops immediately after the current instruction has been completely executed (post instruction).
- After enabling or disabling a trace source, the change will be visible at the next instruction that is executed.
- Quantum end detection is as for syncLevel 2. This allows switching between syncLevels 2 and 3 without changing the simulation scheduling.
- Use cases: a `stop` from within arbitrary MTI callbacks such as the `INST` callback. This syncLevel is a fallback for all use cases that do not fall into syncLevels 0-2.

1.8.2 Controlling and observing the syncLevel

CADI watchpoints automatically register and unregister for their required syncLevel (`POST_INSN_IO`). All other use cases must explicitly register and unregister for the syncLevel they require.

Users of syncLevel write to a set of non-architectural processor registers in the CADI and SCADI interface to register and unregister for specific syncLevels. Processor registers are more suitable than CADI parameters for exposing an interface that has side effects on writes and where values might change spontaneously.

This is the exposed interface to control and observe the value of syncLevel. All these registers are in the CADI/SCADI register group `simulation` for each CT processor that contains non-architectural, simulator-specific registers. All are 32-bit integer registers. Users of syncLevel write to these registers to register and unregister for the syncLevel they require:

syncLevelSyncStateRegister

Users write to this register for `SYNC_STATE`. Write-only.

syncLevelSyncStateUnregister

Users write this to unregister for `SYNC_STATE`. Write-only.

syncLevelPostInsnIORegister

Users write to this register for `POST_INSN_LDST`. Write-only.

syncLevelPostInsnIOUnregister

Users write this to unregister for `POST_INSN_LDST`. Write-only.

syncLevelPostInsnAllRegister

Users write this to register for `POST_INSN_ALL`. Write-only.

syncLevelPostInsnAllUnregister

Users write this to unregister for `POST_INSN_ALL`. Write-only.

These registers are only for debugging and visibility in the debugger, and syncLevel users do not usually access them at all:

syncLevel

Current syncLevel. Read-only.

syncLevelSyncStateCount

User counter. Read/write (use as read-only).

syncLevelPostInsnIOCount

User counter. Read/write (use as read-only).

syncLevelPostInsnAllCount

User counter. Read/write (use as read-only).

minSyncLevel

Same as the `min_sync_level` parameter, described below. Read/write.

The `syncLevelxxxRegister` and `syncLevelxxxUnregister` registers are for syncLevel users to register and unregister themselves, by writing the value 0 to them. Changes to the syncLevel become effective at the next stop event checkpoint. In addition, syncLevel users can write to these registers any time before the simulation is running, for example from the `init()` simulation phase. The change takes effect immediately when the simulation is run.

The other registers are only present to make the debugging of these mechanisms and their users easier. The `syncLevel` register enables you to see what kind of performance you can expect from the model. You must treat access to these other registers as read-only. You can write to them, however, to permit debugging the syncLevel mechanisms.

These registers are not memory (or CPnn-) mapped anywhere, and are not accessible to target programs.

In addition to this debug register interface, there is a CADI parameter that can influence the syncLevel:

```
min_sync_level (default=0, type=int, runtime=true)
```

This parameter enables you to control the minimum syncLevel by the CADI parameter interface. This is not intended to be the primary interface to control the syncLevel because it does not enable multiple independent syncLevel users to indicate their requirements to the simulator. It is primarily for debugging purposes and for situations where a single global syncLevel setting is sufficient.

You can change this parameter at runtime, and changes become effective at the next stop event checkpoint. Reading this parameter value returns the `min_sync_level`, not the current syncLevel. This parameter is only an additional way of controlling the syncLevel and controls the same mechanisms as the register interface.

1.9 SCADI

A Synchronous CADI (SCADI) interface enables you to request a synchronous stop of the simulation. It offers direct synchronous, but unsynchronized, register and memory read access.

The intention is that this interface is used by code that is inherently synchronized with the simulation, in particular from code that is part of the simulation itself (simulation thread). Examples of code that could use this interface are SystemC and LISA peripheral device or bus access functions, and MTI callback functions that are always called synchronously by the simulation itself.



The SCADI interface is not a replacement for CADI. It is an additional interface with clear semantics that differ slightly from CADI.

1.9.1 Intended uses of CADI and SCADI

Component Architecture Debug Interface (CADI) and Synchronous CADI (SCADI) have separate design principles.

CADI

Used by debuggers and simulation controllers to control the simulation. Execution control functions are always asynchronous and are usually called from a non-simulation thread, usually the debugger GUI thread. You can also use CADI to read/write registers, memory and so on, but only from non-simulation threads, for example only from the debugger thread.

SCADI

Implements a specific subset of CADI functions, that is, mainly read/write registers and memory, set and clear breakpoints, and get disassembly. You can only use SCADI from the simulation thread itself and from threads that can make sure on their own that the simulation is currently blocked, for example a debugger thread while it is sure that the simulation is in a stable state. SCADI is intended to be used from within peripheral read/write accesses while the simulation is running, or from within MTI callbacks that the simulation is running.

SCADI does not implement and execute control functions except `CADIExecStop()`, because only stop makes sense from within the simulation thread. Asynchronous functions such as `CADIExecContinue()` and `CADIExecSingleStep()` are not supported by SCADI, so for these you must use CADI instead.

The `SCADI::CADIExecStop()` function is the exception, because for CADI this means “stop when it is convenient for the simulation”, which is asynchronous, but `CADIExecStop()` is synchronous and means “stop now”, which you enable with the appropriate `syncLevel >= 2`.

The guidelines are:

- Use CADI for execution control.
- Use CADI or SCADI for read/write registers and memory, depending on the situation:

- Use SCADI if the caller can make sure that the accesses are (potentially inherently) synchronized with the simulation.
- Use CADI if called from a debugger thread that does not know exactly whether the simulation is running or is in a stable state.

1.9.2 Responsibilities of the SCADI caller

Synchronous interface means that the caller of SCADI functions is always responsible for ensuring that the simulation is in a stable state.

Being in a stable state means that the simulation does not advance while the call is being made. The caller is also responsible for meeting all host thread and synchronization requirements of any external bus slaves that are directly or indirectly connected to the memory bus.

1.9.3 SCADI interface access

You can obtain the SCADI interface from the same `CAInterface` pointers that provide the CADI and MTI interfaces.

This is done using the `CAInterface::ObtainInterface()` method. The interface names for the normal CADI and MTI interfaces are `CADI::IFNAME()` and `ComponentTraceInterface::IFNAME()`, respectively.

The interface name for the SCADI interface to pass into `ObtainInterface()` is `"eslapi.SCADI2"` and the interface revision is 0. There are no `IFNAME()` and `IFREVISION()` static functions defined for SCADI, and a plain string constant and integer constant (0) are used instead.

The pointer returned by `CAInterface::ObtainInterface("eslapi.SCADI2")` must be cast into an `eslapi::CADI` class pointer, that is, the same class as for `eslapi.CADI2`, the normal CADI interface class.

1.9.3.1 Access to SCADI from within LISA components

To access the SCADI interface of the LISA component itself, use the `getSCADI()` function. This returns the SCADI interface of the LISA component itself.

To access the SCADI interfaces of other components in the system, use the `getGlobalInterface()` function. This returns a `CAInterface` pointer to a simulation-wide Component Registry.

You then have to use this registry pointer with the `obtainComponentInterfacePointer()` function to obtain the `SystemTraceInterface`. This interface provides a list of all components in the system that provide trace data. You can use this list to access the SCADI interfaces of all the components.

1.9.3.2 Access to SCADI from within SystemC

To access the SCADI interfaces from within SystemC, use the `scx_get_global_interface()` function. This returns a `CAInterface` pointer to a simulation-wide Component Registry.

You then have to use this registry pointer with the `obtainComponentInterfacePointer()` function to obtain the `SystemTraceInterface`. This interface provides a list of all components in the system that provide trace data. You can use this list to access the SCADI interfaces of all the components.

1.9.3.3 Access to SCADI from MTI plug-ins

MTI plug-ins can gain access to SCADI interfaces by using the `CAInterface` pointer, which is passed as a parameter to the `RegisterSimulation()` function.

This pointer has the same semantics as the `CAInterface` pointer returned by the `getGlobalInterface()` functions in the SystemC or LISA use cases.

1.9.3.4 Example code for retrieving a SCADI interface pointer of a component

This code demonstrates how to use `getGlobalInterface()` to retrieve a SCADI interface pointer of a specific component.

From within LISA, use `getGlobalInterface()`. From within SystemC, it is assumed that you have a pointer to the Fast Models platform called `platform`. From within an MTI plug-in, you do not have to call `getGlobalInterface()`, and you can use the `CAInterface` pointer passed to `RegisterSimulation()`.

```
#include "sg/SGComponentRegistry.h"
eslapi::CADI *Peripheral::get_SCADI_interface(const char *instanceName)
{
    // get access to MTI interface (SystemTraceInterface)
    // - this code is for a SystemC peripheral
    // - for LISA this would be just 'getGlobalInterface()'
    // - for MTI plugins this would be just using the CAInterface *caif pointer passed
    // into RegisterSimulation()
    eslapi::CAInterface *globalInterface = scx::scx_get_global_interface();
    const char *errorMessage = "(no error)";
    MTI::SystemTraceInterface *mti = 0;
    if (globalInterface)
        mti = sg::obtainComponentInterfacePointer<MTI::SystemTraceInterface>
            (globalInterface,
             "mtiRegistry",
             &errorMessage);

    if (mti == 0)
    {
        printf("ERROR:MTI interface not found! (%s)\n", errorMessage);
        return 0;
    }
    // find component with instanceName
    eslapi::CAInterface *compif = 0;
    for (MTI::SystemTraceInterface::TraceComponentIndex i = 0;
         i < mti->GetNumOfTraceComponents();
         i++)
    {
        const char *name = mti->GetComponentTracePath(i);
        if (verbose)
```

```

        printf("TEST MTI component '%s'\n", name);
        if (strcmp(name, instanceName) == 0)
            compif = mti->GetComponentTrace(i);
    }
    if (!compif)
    {
        printf("ERROR:instance '%s' not found while trying to get SCADI! \n",
               instanceName);
        return 0;
    }
    // get and return SCADI interface
    return static_cast<eslapi::CADI *>(compif->ObtainInterface("eslapi.SCADI2", 0,
0));
}

```

1.9.4 SCADI semantics

This section lists the available subset of functionality and the differences in semantics from the CADI interface.

The SCADI interface provides a subset of functionality of the CADI interface. All functions in SCADI differ from the CADI functions, that is, the caller is responsible for satisfying the constraints and requirements in terms of synchronization with the simulation.

These functions have the same semantics as in CADI:

- CADIRegGetGroups()
- CADIRegGetMap()
- CADIRegGetCompount()
- CADIMemGetSpaces()
- CADIMemGetBlocks()
- CADIMemGetOverlays()
- CADIGetParameters()
- CADIGetParameterInfo()
- CADIXfaceGetFeatures()

The following debug read access functions have the same semantics as in CADI, but the values that these read access functions return might differ from the values returned by the corresponding CADI functions because some of the accessed resource values might change during the execution of an instruction.

The values of resources that are modified by the current instruction are **UNKNOWN**, except the PC, which always reflects the address of the last instruction that was issued. Reading and writing registers and memory while the simulation is running is generally only useful for `syncLevel >= SL_SYNC_STATE`. For `SL_OFF`, all registers and all memory locations are always **UNKNOWN** while the simulation is running.

- CADIRegRead()
- CADIMemRead()

- `CADIGetParameterValues()`
- `CADIGetInstructionCount()`
- `CADIGetPC()`
- `CADIGetDisassembler()`

The following write access functions always provide write access to the register in the `simulation` register group. They also might provide limited write access semantics to certain resources. A SCADI implementation might choose not to support any write access to core registers and memory:

- `CADIRegWrite()`
- `CADIMemWrite()`
- `CADISetParameters()`

The breakpoint functions have nearly all the same semantics as in the CADI interface, except that changes to breakpoints only become effective at the next stop event checkpoint defined by the current `syncLevel`. Changes might also only become visible in `CADIBptRead()` and `CADIBptGetList()` at this next synchronization point.

- `CADIBptGetList()`
- `CADIBptRead()`
- `CADIBptSet()`
- `CADIBptClear()`
- `CADIBptConfigure()`

Execution control is limited to stopping the simulation. The simulation stops at the next synchronization point as defined by the current `syncLevel`:

- `CADIExecStop()`.

All other functions are unsupported, and return `CADI_STATUS_CmdNotSupported`.

1.10 User mode networking

User mode networking emulates a built-in IP router and DHCP server, and routes TCP and UDP traffic between the guest and host. It uses the user mode socket layer of the host to communicate with other hosts.

This allows the use of a significant number of IP network services without requiring administrative privileges, or the installation of a separate driver on the host on which the model is running. Fast Models supports the following kinds of Ethernet device models:

SMSC_91C111

This is paired with an external `HostBridge` component. The user mode networking specification is set on the external `HostBridge`.

VirtioNetMMIO

This has a built-in HostBridge sub-component. The user mode networking specification is set on the internal HostBridge.



Note

- You can use TCP and UDP over IP, but not ICMP (ping).
- User mode networking does not support forwarding UDP ports on the host to the model.
- You can only use DHCP within the private network.
- You can only make inward connections by mapping TCP ports on the host to the model. This is common to all implementations that provide host connectivity using NAT.
- Operations that require privileged source ports, for example NFS in its default configuration, do not work.
- If setup fails, or the parameter syntax is incorrect, there is no error reporting.

To enable user mode networking, run the model with the following CADI parameters to activate the components:

SMSC_91C111:

```
-C motherboard.hostbridge.userNetworking=true
-C motherboard.smc_91c111.enabled=true
```

VirtioNetMMIO:

```
-C motherboard.virtio_net.hostbridge.userNetworking=true
-C motherboard.virtio_net.enabled=true
```

To map a host TCP port to a model port, run the model with the `userNetPorts` parameter. This parameter allows services to appear to be listening on privileged ports in the model but be mapped to unprivileged ports on the host. The syntax is a comma-separated list of items in the form:

```
[host-ip:]hostport=[model-ip:]modelport
```

For example, to map port 8022 on the host to port 22 on the model, use this parameter:

SMSC_91C111:

```
-C motherboard.hostbridge.userNetPorts=8022=22
```

VirtioNetMMIO:

```
-C motherboard.virtio_net.hostbridge.userNetPorts=8022=22
```

Either or both of a host IP address and model IP address can optionally be specified on either side of the assignment to select a specific interface on which the mapping will occur. For example:

```
127.0.0.1:8022=127.0.0.1:22
```

The default is to accept connections on any interface.

Related information

- [HostBridge](#)

1.11 TAP/TUN networking

This section describes TAP/TUN networking.

1.11.1 TAP/TUN networking limitations

This section describes general limitations of TAP/TUN networking for models.

DHCP

If the host uses Dynamic DNS, it inserts records into DNS. If you manage this host with DHCP, then installing TAP networking can cause failure to register in the DNS. After the physical device attaches to the bridge device, the DHCP client reruns, but the DHCP request does not have the correct hostname.

WiFi

Most WiFi adaptors do not implement the required support for TAP networking to work.

1.11.2 Setting up a network connection for Microsoft Windows

This section describes how to set up a network connection.

Before you begin

The Fast Models networking environment needs the Third Party IP add-on for the Fast Model Portfolio.

Procedure

1. Close all non-essential applications.
2. Install the TAP driver and configure it:
 - a) Locate the Fast Models Portfolio, by default `C:\Program Files\ARM\FastModelsPortfolio_X.Y\`.
 - b) Execute the `ModelNetworking\add_adapter_64.bat` file.
3. Select **Start > Control Panel > Network Connections** and locate the newly installed TAP device.
4. Select at least one real Ethernet adapter connection. Press the **Ctrl** key to multi-select.
5. Right-click and select **Bridge Connections**.

1.11.3 Configuring the networking environment for Microsoft Windows

This section describes how to set the parameters to make a network connection.

About this task



- This procedure has not been tested with wireless network adapters.
- Firewall software might block network traffic in the network bridge, and might result in a networking failure in the model. If the model does not work after configuration, check the firewall settings.

Procedure

1. Set the appropriate parameters on the `HostBridge` and `SMSC_91C111` components, or on the `VirtioNetMMIO` component and its `HostBridge` subcomponent. For example:

SMSC_91C111:

```
hostbridge.interfaceName="ARM<x>"
smc_91c111.enabled=1
```

VirtioNetMMIO:

```
virtio_net.hostbridge.interfaceName="ARM<x>"
virtio_net.enabled=1
```

`ARM<x>` is an adapter that is built into the network bridge, and is 0 by default. Use `x+1` if `ARM<x>` exists.

2. Set another parameter on the device, if you have to enable promiscuous mode for the TAP device.
 - a. Select **Start > Run > cmd.exe**.
 - b. Enter the following commands.

```
netsh bridge show adapter
netsh bridge set adapter <N> forcecompatmode=enable
```

`<N>` is the id number of the TAP adapter that the first command lists.

1.11.4 Commands for `tap_setup_32.exe` or `tap_setup_64.exe` for Microsoft Windows

This section lists the commands for use with `tap_setup_32.exe` or `tap_setup_64.exe`. These files are located in the `ModelNetworking` directory.

help

Print help information.

set_media

Configure the TAP devices to `Always Connected`.

set_perm

Configure the TAP devices to `Non-admin Accessible`.

restart

Restart the TAP devices.

list_dev

List available TAP devices and output to a file.

rename

Rename the device to **ARMx**.

install <infile> <id>

Install a TAP Win32 or Win64 adapter.

remove <dev>

Remove TAP Win32 or Win64 adapters. Set <dev> to `All` to remove all tap devices.

setup <infile> <id>

Automated setup process.

1.11.5 Uninstalling networking for Microsoft Windows

This section describes how to uninstall TAP adapters from the base location of the Fast Models Portfolio, by default `C:\Program Files\ARM\FastModelsPortfolio_X.Y`.

Procedure

1. Run the `remove_all_adapters_32.bat` Or `remove_all_adapters_64.bat` file. These files are located in the `ModelNetworking` directory.
2. If the uninstallation does not delete the bridge, delete it manually:
 - a) Close all non-essential applications.
 - b) Select **Network Connections**.
 - c) Right-click on the bridge and select **Delete**.

1.11.6 Setting up a network connection for Red Hat Enterprise Linux

This section describes how to set up a network connection.

Before you begin

Ensure that the `brctl` utility is on your system. This utility is in the installation package, but we recommend that you use the standard Linux bridge utilities, which are in the Linux distribution.

About this task



- Perform this procedure once for each host machine.
 - The setup and configuration instructions assume that your network provides IP addresses by DHCP. Otherwise, consult your network administrator.
-

Procedure

1. In a shell, change to the `FastModelsPortfolio_<X.Y>/ModelNetworking` directory.
2. Run the following script from this directory, because it does not work correctly if run from any other location:

32-bit operating system

Run `add_adapter_32.sh` as root. For example, `sudo ./add_adapter_32.sh`.

64-bit operating system

Run `add_adapter_64.sh` as root. For example, `sudo ./add_adapter_64.sh`.

3. The prompt appears: **Specify the TAP device prefix:(ARM)**. Select **Enter** to accept the default.
 4. The prompt appears: **Specify the user list**. Enter a space-separated list of all users who are to use the model on the network, then select **Enter**. All entries in the list must be the names of existing user accounts on the host.
 5. The prompt appears: **Enter the network adapter which connects to the network:(eth0)**. Select **Enter** to accept the default, or input the name of a network adapter that connects to your network.
 6. The prompt appears: **Enter a name for the network bridge to create:(armbr0)**. Select **Enter** to accept the default, or input a name for the network bridge. You must not have an existing network interface on your system with the selected name.
 7. The prompt appears: **Enter the location to write the init script to:(/etc/init.d/FMNetwork)**. Select **Enter** to accept the default, or input another path with a new filename in an existing directory.
 8. The prompt appears: **WARNING: the script creates a bridge which includes the local network adapter and tap devices. You may suffer temporary network loss. Do you want to proceed? (Yes or No)**. Verify all values input so far, and enter **Yes** if you want to proceed. If you enter **No**, no changes are made to your system.
 9. A prompt appears to inform you of the changes that the script is to make to your system. Input **Yes** if you are happy to accept these changes, or input **No** to leave your system unchanged.
-



After entering **Yes**, you might temporarily lose network connectivity. Also, the IP address of the system might change.

Next steps

The network bridge is disabled after the host system is reset. To configure the host system to support bridged networking, you might have to create links to the `init` script (FMNetwork). The script suggests some appropriate links for Red Hat Enterprise Linux.

The default firewall configuration on Red Hat Enterprise Linux blocks packet transmission across the TAP networking bridge device. You can disable the firewall. If the context makes this unwise, then add firewall rules to allow transmission. These `iptables` commands configure the firewall to allow packets across the bridge device:

```
iptables -I FORWARD -m physdev --physdev-is-bridged -j ACCEPT
service iptables save
service iptables restart
```

1.11.7 Setting up a network connection for Ubuntu Linux

This section describes how to set up a network connection.

About this task

To use TAP networking with Fast Models on Ubuntu, set up a TAP device manually by following these steps. This guide uses a network interface `eth0` and a username `fmuser`. Replace these values as appropriate.



Typographic errors when modifying the network configuration can cause failure to connect to the network. We recommend performing these steps on a machine that you have physical access to.

Procedure

1. If it is not present, add `eth0` to the interfaces file `/etc/network/interfaces`. This step stops network-manager from managing `eth0`. It can result in network-manager indicating there is no network connection even if there is. You must have root privileges for this step.

Use one of the following ways:

- For an interface using DHCP, add:

```
auto eth0
iface eth0 inet dhcp
```

- To configure a static IP address, add the static information, for example:

```
auto eth0
iface eth0 inet static
address 192.168.0.2
netmask 255.255.255.0
gateway 192.168.0.1
```

The network notifier applet launches the GUI tool network-manager. It automatically configures network devices that `/etc/network/interfaces` does not manage, and sets up devices in a way that is incompatible with bridging. This step ensures that network-manager does not manage the network interface that you want to bridge to. If you are unsure how to configure your network interface, ask your network administrator.

2. Install the bridge-utils package:

```
sudo apt-get install bridge-utils
```

This step provides the `brctl` command for creating and managing the network bridge.

3. Create a bridge device by adding this entry to `/etc/network/interfaces`:

```
auto armbr0
iface armbr0 inet dhcp
pre-up ifconfig eth0 0.0.0.0 promisc
post-down ifconfig eth0 0.0.0.0 -promisc
```

The pre-up and post-down lines give commands to execute before bringing up `armbr0` and after bringing it down. These commands put `eth0` into promiscuous mode at pre-up and take it out of promiscuous mode at post-down. Promiscuous mode makes sure that the hardware does not filter out packets for the virtual ethernet device.

This step creates a bridge device that is called `armbr0` from Fast Models TAP devices to the physical network.

4. Create the TAP devices. TAP devices need permission for specific users, so create one for each user who is to run the model with the virtual ethernet device.
For example, to create a TAP device called `ARMfmuser` for the user `fmuser`, add the following lines to the `armbr0` section of `/etc/network/interfaces`.

```
pre-up ip tuntap add dev ARMfmuser mode tap user fmuser
pre-up ifconfig ARMfmuser 0.0.0.0 promisc
post-down ip tuntap del dev ARMfmuser mode tap
```

This step creates a TAP device for each user.

5. Create a bridge between the TAP devices and the network interface `eth0` by adding a `bridge_ports` line to the `armbr0` section of `/etc/network/interfaces`. For example, for a TAP device that is named `ARMfmuser`, add the following line:

```
bridge_ports eth0 ARMfmuser
```

6. The added `/etc/network/interfaces` code now looks like this:

```
auto armbr0
iface armbr0 inet dhcp
pre-up ifconfig eth0 0.0.0.0 promisc
post-down ifconfig eth0 0.0.0.0 -promisc
pre-up ip tuntap add dev ARMfmuser mode tap user fmuser
pre-up ifconfig ARMfmuser 0.0.0.0 promisc
post-down ip tuntap del dev ARMfmuser mode tap
bridge_ports eth0 ARMfmuser
```


- Restart network services by either restarting the computer or by running the following commands:

```
sudo ifdown eth0 && sudo ifup eth0
sudo ifup armbr0
sudo service network-manager restart
```



armbr0 must be explicitly started.

This step disconnects and reconnects all network interfaces.

1.11.8 Configuring the networking environment for Linux

This section describes how to set the parameters to make a network connection.

Before you begin

Use System Canvas or a related Fast Models tool to load a project or model, and then select a component.

About this task



Firewall software might block network traffic in the network bridge, and result in a networking failure. If the model does not work after configuration, check the firewall settings.

Procedure

Set the parameters on the `HostBridge` and `SMSC_91C111` components, or on the `virtioNetMMIO` component and its `HostBridge` subcomponent. For example:

SMSC_91C111:

```
hostbridge.interfaceName=ARM<username>
smc_91c111.enabled=1
```

VirtioNetMMIO:

```
virtio_net.hostbridge.interfaceName=ARM<username>
virtio_net.enabled=1
```

ARM<username> is an adapter that is built into the network bridge.

Related information

[Fast Models Tools User Guide](#)

1.11.9 Solutions to networking issues on Linux

This section describes how to solve networking issues.

The model networking works after initial setup, but stops working after reboot

Set the correct access permissions for the `/dev/net/tun` device, by executing `chmod 666 /dev/net/tun` as root. To preserve the change across reboots, modify the udev rules of the TAP device by opening `/etc/udev/rules.d/50-udev.rules` as root, and finding the line:

```
KERNEL=="tun", NAME="net/%k"
```

If it does not have `MODE="0666"` at the end of the line, append `MODE="0666"`:

```
KERNEL=="tun", NAME="net/%k", MODE="0666"
```

Model networking installs correctly, but when a model starts up, the model cannot receive packets

Disable the firewall on the host machine, or add the TAP device to `trusted devices`.



Refer to the vendor's documentation manual.

1.11.10 Disabling and re-enabling networking for Linux

This section describes how to disable and re-enable networking with an `init` script.

About this task



These operations remove/restore TAP devices and the network bridge. There is a temporary loss of network connectivity and your IP address might change.

Procedure

1. To disable networking without uninstalling it, invoke the installed `init` script (by default, `/etc/init.d/FMNetwork`) as root with the parameter `stop`:

```
sudo /etc/init.d/FMNetwork stop
```

2. To re-enable networking, invoke the `init` script as root with the parameter `start`:

```
sudo /etc/init.d/FMNetwork start
```

1.11.11 Uninstalling networking for Linux

This section describes the steps to uninstall a network.

Procedure

1. In a shell, change to the `/FastModelsPortfolio_X.Y/ModelNetworking/` directory.
2. Run `uninstall.sh` as root, passing the location of the `init` script (FMNetwork):

```
sudo ./uninstall.sh /etc/init.d/FMNetwork
```

You must run this script from the directory in which it is installed, because it does not work correctly if run from any other location.



There is a temporary loss of network connectivity and your IP address might change.

Next steps

The uninstall script removes everything that can be safely removed. It does not remove:

- symlinks to the `init` script. You must remove any symlinks that you have created.
- `/sbin/brcctl`. Removing this is optional.

1.12 Using parameters to set port values

Some processor and peripheral component ports are almost always static in value when used as part of a typical platform. For example, the reset vector base address register address (RVBARADDR) port in processor components.

To facilitate easy configuration of platform models, the IP models for these components can provide a shadow parameter for these ports. This parameter can be used to change the value that is used by the model. In these cases, the following rules apply:

- If a port is driven in the platform model, then the parameter value is ignored.
- If a port is not driven in the platform model, then the parameter value is sampled at both simulator reset, and at every subsequent simulation reset of the specific IP model.



Simulator reset corresponds with the LISA `reset()` behavior and the SystemC `start_of_simulation()` callback.

- All ports and parameters that are sampled at reset are sampled when the simulation reset signal concerned is deasserted.

- If a port is not driven in the platform model, and a parameter has not been set, then the default value for the parameter is used.

In some IP models, the value of some ports can only be set by using a parameter. That is, the parameter is provided instead of the port.

1.13 PVBUS C++ transaction and Tx_Result classes

This section describes the C++ transaction and `Tx_Result` classes.

1.13.1 Class `pv::TransactionGenerator`

This class provides efficient mechanisms for bus masters to generate transactions that are transmitted over the `pvbustm` port of the associated PVBUSMaster subcomponent.

You can produce `pv::TransactionGenerator` objects by invoking the `createTransactionGenerator()` method on the control port of a PVBUSMaster component.

```
class pv::TransactionGenerator
{
    // Tidy up when TransactionGenerator is deleted.
    ~TransactionGenerator()

    // Control AXI-specific signal generation for future transactions.
    // Privileged processing mode.
    void setPrivileged(bool priv = true);

    // Instruction access (vs data).
    void setInstruction(bool instr = true);

    // Normal-world access (vs secure).
    void setNonSecure(bool ns = true);

    // Locked atomic access.
    void setLocked(bool locked = true);

    // Exclusive atomic access.
    void setExclusive(bool excl = true);

    // Generate transactions.
    // Generate a read transaction.
    bool read(bus_addr_t, pv::AccessWidth width, uint32_t *data);

    // Generate a write transaction.
    bool write(bus_addr_t, pv::AccessWidth width, uint32_t const *data);

    // Generate read transactions.
    bool read8(bus_addr_t, uint8_t *data);
    bool read16(bus_addr_t, uint16_t *data);
    bool read32(bus_addr_t, uint32_t *data);
    bool read64(bus_addr_t, uint64_t *data);

    // Generate write transactions.
    bool write8(bus_addr_t, uint8_t const *data);
    bool write16(bus_addr_t, uint16_t const *data);
    bool write32(bus_addr_t, uint32_t const *data);
    bool write64(bus_addr_t, uint64_t const *data);
};
```

1.13.2 TransactionGenerator efficiency considerations

TransactionGenerators are most efficient for multiple accesses to one 4KB page.

Each TransactionGenerator caches connection information internally. This improves efficiency for multiple accesses to a single 4KB page. If a component requires repeated access data from different pages, for example when streaming from one location to another, we recommend you create a TransactionGenerator for each location.

You can dynamically create and destroy TransactionGenerators, but it is better to allocate them once at initialization and destroy them at shutdown. See the example in `$PVLIB_HOME/examples/LISA/BusComponents/DmaTransfer.lisa`.

1.13.3 Enum `pv::AccessWidth`

This enum selects the required bus width for a transaction.

Defined values are:

- `pv::ACCESS_8_BITS`
- `pv::ACCESS_16_BITS`
- `pv::ACCESS_32_BITS`
- `pv::ACCESS_64_BITS`

1.13.4 Class `pv::Transaction`

This class is a base class for read and write transactions that are visible in the PVBUSSlave subcomponent. It contains functionality common to both types of transaction.

This class provides an interface that permits bus slaves to access the details of the transaction. Do not instantiate these classes manually. The classes are generated internally by the PVBUS infrastructure.

This base class provides access methods to get the transaction address, access width, and bus signals. It also provides a method to signal that the transaction has been aborted.

```

class pv::Transaction
{
public:
    // Accessors
    bus_addr_t getAddress() const;           // Transaction address.
    pv::AccessWidth getAccessWidth() const; // Request width.
    int getAccessByteWidth() const;          // Request width in bytes.
    int getAccessBitWidth() const;           // Request width in bits.

    bool isPrivileged() const; // Privileged process mode?
    bool isInstruction() const; // Instruction request vs data?
    bool isNonSecure() const;  // Normal-world vs secure-world?
    bool isLocked() const;     // Atomic locked access?
    bool isExclusive() const;  // Atomic exclusive access?

```

```

uint32_t getMasterID() const;
bool hasSideEffect() const;

// Generate transaction returns
Tx_Result generateAbort();           // Cause the transaction to abort.
Tx_Result generateSlaveAbort();       // Cause the transaction to abort.
Tx_Result generateDecodeAbort();     // Cause the transaction to abort.
Tx_Result generateExclusiveAbort();  // Cause the transaction to abort.
Tx_Result generateIgnore();
};

```

1.13.5 Class pv::ReadTransaction

This class extends the pv::Transaction class to provide methods for returning data from a bus read request.

```

class ReadTransaction : public Transaction
{
public:
    /*! Return a 64-bit value on the bus. */
    Tx_Result setReturnData64(uint64_t);

    /*! Return a 32-bit value on the bus. */
    Tx_Result setReturnData32(uint32_t);

    /*! Return a 16-bit value on the bus. */
    Tx_Result setReturnData16(uint16_t);

    /*! Return an 8-bit value on the bus. */
    Tx_Result setReturnData8(uint8_t);

    /*! This method provides an alternative way of returning a Tx_Result
     * success value (instead of just using the value returned from
     * setReturnData<n>()).
     *
     * This method can only be called if one of the setReturnData<n>
     * methods has already been called for the current transaction.
     */
    Tx_Result readComplete();
};

```

1.13.6 Class pv::WriteTransaction

This class extends the pv::Transaction class to provide methods for returning data from a bus write request.

```

class WriteTransaction : public Transaction
{
public:
    /*! Get bottom 64-bits of data from the bus. If the transaction width
     * is less than 64-bits, the data is extended as appropriate.
     */
    uint64_t getData64() const;

    /*! Get bottom 32-bits of data from the bus. If the transaction width
     * is less than 32-bits, the data is extended as appropriate.
     */
    uint32_t getData32() const;

    /*! Get bottom 16-bits of data from the bus. If the transaction width

```

```
* is less than 16-bits, the data is extended as appropriate.
*/
uint16_t getData16() const;

/*! Get bottom 8-bits of data from the bus. If the transaction width
* is less than 8-bits, the data is extended as appropriate.
*/
uint8_t getData8() const;

/*! Signal that the slave has handled the write successfully.
*/
Tx_Result writeComplete();
};
```

1.14 Visualisation library

The Visualisation library does not model hardware directly but instead provides components, protocols, and a library. These permit a GUI display that lets you interact with the external I/O from the model platform.

The types of I/O handled include:

- LCD display, such as the output from the PL110_CLCD component display port.
- LEDs representing values from a ValueState port as either single lights, or as segmented alphanumeric displays.
- DIP switches, which can drive a ValueState port.
- Capture of keyboard and mouse input, using the KeyboardStatus and MouseStatus protocols to feed input to a PS2Keyboard or PS2Mouse component.
- Background graphics, custom rendered graphics, and clickable push buttons, permitting the UI to display a skin representing the physical appearance of the device being modeled.
- Status information such as processor instruction counters, with values taken from the InstructionCount port of a processor.

The Visualisation library provides a C++ API that enables you to write your own visualization components in LISA+. These custom components can display any combination of the supported I/O types.

You can add the prebuilt GUIPoll component to your custom component. The GUIPoll component provides a LISA visualization component with a periodic signal that keeps the display updated, even when the simulation is stopped.

The Visualisation library supports one signaling protocol, the LCD protocol.

Related information

[LCD protocol](#) on page 80

[LISA visualisation models](#) on page 59

[Visualisation library C++ classes](#) on page 60

1.14.1 LISA visualisation models

The visualisation components provide a host window to display status information in addition to a frame buffer.

Each example platform model contains its own LISA visualisation model. You can use the model as the basis for your own visualization-containing components, such as the PL110_CLCD component. To use the visualisation components in your own system, copy the component LISA files from the relevant platform model directory, because they are not in the generic model library.

Related information

[Versatile Express model](#) on page 4260

1.14.2 Visualisation library C++ classes

This section describes the C++ classes and structures in the Visualisation library.

1.14.2.1 C++ classes inclusion

To use these Visualisation library classes, begin your LISA component with the correct `#include` statement.

```
includes
{
#include "components/Visualisation.h"
}
```

1.14.2.2 Class Visualisation

The `visualisation` class is the API for creating a custom LISA visualization component.

A component obtains an instance of this class by calling the global function `createVisualisation()`. The component can then use this instance to control the size and layout of the visualization window:

Visualisation *createVisualisation()

This function generates an instance of the Visualisation library. You can only call this function once, because SDL only supports opening a single display window. The Visualisation library is implemented using the Simple DirectMedia Layer (SDL) cross-platform rendering library.

The `visualisation` class has the following methods:

~Visualisation()

Destructor for the Visualisation library. You must only call this method when your simulation is shutting down, after all allocated resources (`VisRenderRegions`, `VisPushButtonRegions`, `VisBitmaps`) have been deleted.

void configureWindow(unsigned int width, unsigned int height, unsigned int bit_depth)

Sets the visualization window to the requested size and bit depth. Depending on the display capabilities, the window might actually get a different bit depth from the size you requested.

VisBitmap *loadImage(char const *filename)

Allocates a new VisBitmap object, initialized by loading a Microsoft Windows Bitmap (.BMP) from the given file.

VisBitmap *loadImageWithAlphaKey(char const *filename, unsigned int red, unsigned int green, unsigned int blue)

Allocates a VisBitmap object, as with `loadImage()`. All pixels of the color specified by `red`, `green`, `blue` are converted into a transparent alpha channel.

VisBitmap *cropImage(VisBitmap *source, int x, int y, unsigned int width, unsigned int height)

Allocates a new VisBitmap object, by cropping a region from the source bitmap.

void releaseImage(VisBitmap *)

Releases the resources held by the given VisBitmap. The underlying bitmap is only to be unloaded if it is not in use.

void setBackground(VisBitmap *background, int x, int y)

Sets the background image for the visualization window. This takes a copy of the data referenced by the VisBitmap, so it is safe for the client to call `releaseImage(background)` immediately after calling `setBackground()`. The background is not displayed until the first call to `poll()`.

VisRenderRegion *createRenderRegion()

Allocates a new VisRenderRegion object that can be used to display arbitrary graphics, including LCD contents, in a rectangular region.

VisPushButtonRegion *createPushButtonRegion()

Allocates a new VisPushButtonRegion, which can be placed at a location on the display to provide a clickable push button.

bool poll(VisEvent *event)

Permits the Visualisation library to poll for GUI events. The client passes a reference to a VisEvent structure, which receives details of a single mouse/keyboard event.

The method returns false if no events have occurred.

Your LISA visualization implementations must call this periodically by using a GUIPoll component. On each `gui_callback()` event, you must ensure that the visualization component repeatedly calls `poll()` until it returns false.

void lockMouse(VisRegion *region)

Locks the mouse to the visualization window and hides the mouse pointer.

void unlockMouse()

Unlocks and redisplay the mouse pointer.

bool hasQuit()

Returns true if the user has clicked on the close icon of the visualization window.

1.14.2.3 Class VisRegion

This class is the common base class for VisPushButtonRegion and VisRenderRegion, representing a region of the visualization display.

~VisRegion()

Permits clients to delete a VisPushButtonRegion when it is no longer required.

void setId(void *id)

Permits a client-defined identifier to be associated with the region.

void *getId()

Returns the client-defined identifier.

void setVisible(bool vis)

Specifies whether the region is to be displayed on the screen. This is currently ignored by the SDL implementation.

void setLocation(intx, int y, unsigned int width, unsigned int height)

Sets the location of this region relative to the visualization window.

1.14.2.4 Class VisPushButtonRegion : public VisRegion

This class defines a region of the visualization window that represents a clickable button.

Optionally, the button can provide different VisBitmap representations for a button-up and a button-down graphic, and a graphic to use when the mouse pointer rolls over the button.

In addition to the public method defined in VisRegion, this class defines these methods:

- `void setButtonUpImage(VisBitmap*bmpUp) : void`
- `setButtonDownImage(VisBitmap*bmpDown) : void`
- `setButtonRollOverImage(VisBitmap*bmpRollover)`

These methods set the graphics to be used for each of the button states. If any image is not specified or is set to NULL, then the corresponding area of the visualization background image is used.

The VisPushButtonRegion takes a copy of the VisBitmap, so the client can safely call `Visualisation::releaseBitmap()` on its copy.

- `void setKeyCode(intcode)`

This method sets the code for the keypress event that is generated when the button is pressed or released.

1.14.2.5 Class VisRenderRegion : public VisRegion

This class defines a region of the visualization window that can render client-drawn graphics, including a representation of the contents of an LCD.

In addition to the public method defined in VisRegion, the class defines these methods:

VisRasterLayout const *lock()

Locks the region for client rendering. While the buffer is locked, the client can modify the pixel data for the buffer. You must not call the methods `writeText()` and `renderBitmap()` while the buffer is locked.

void unlock()

Releases the lock on the render buffer, permitting the buffer to be updated on screen.

void update(int left, int top, unsigned int width, unsigned int height)

Causes the specified rectangle to be drawn to the GUI.

int writeText(const char *text, int x, int y)

Renders the given ASCII text onto an unlocked VisRenderRegion. The return value is the x co-ordinate of the end of the string. The default font is 8 pixels high, and cannot be changed.

void renderBitmap(VisBitmap *bitmap, int x, int y)

Draws a bitmap onto an unlocked VisRenderRegion.

1.14.2.6 Struct VisRasterLayout

This struct defines the layout of the pixel data in a frame-buffer.

The `lock()` method of the LCD protocol expects to be given a pointer to this structure. You can generate a suitable instance by calling `VisRasterRegion::lock()`.

The structure contains these fields:

uint8_t* buffer

This points to the buffer for the rasterized pixel data. The controller can write pixels into this buffer, but must stay within the bounds specified by the width and height.

uint32_t pitch

The number of bytes between consecutive raster lines in the pixel data. This can be greater than the number of bytes per line.

uint32_t width

The width, in pixels, of the render area. This value can be less than the width requested by the LCD controller when it called `lock()`.

uint32_t height

The height, in pixels, of the render area. This value can be less than the height requested by the LCD controller when it called `lock()`.

VisPixelFormat format

This structure defines the format of the pixel data in the raster buffer.

bool changed

This is set to true if the pixel format or buffer size has changed since the previous call to `lock()`.

Pixel data is represented as a one-dimensional array of bytes. The top-left pixel is pointed to by the `buffer` member. Each pixel takes up a number of bytes, given by `format.pbytes`.

The pixel at location (x, y) is stored in the memory bytes starting at:

```
buffer[y * pitch + x * format.pbytes]
```

1.14.2.7 Struct VisPixelFormat

This struct specifies the format of pixel data within the buffer.

The members are:

uint32_trbits, gbits, bbits

The number of bits per color channel.

uint32_t roff, goff, boff

The offset within the pixel data value for the red/green/blue channels.

uint32_t pbytes

The size of a single pixel, in bytes.

`format.pbytes` specifies the number of bytes that make up the data for a single pixel. These bytes represent a single pixel value, stored in host-endian order. The pixel value contains a number of the form:

```
(R<<format.roff) + (G<<format.goff) + (B<<format.boff)
```

where (R,G,B) represents the values of the color channels for the pixel, containing values from 0 up to $(1 \ll \text{format.rbits})$, $(1 \ll \text{format.gbits})$, $(1 \ll \text{format.bbits})$.

2. Protocols

Components communicate through connected ports. Ports have protocols that define the function calls for different connections.

2.1 AMBA-PV protocols

The AMBA-PV components and protocols permit you to model a platform that interfaces with an Arm® AMBA®-based system.

The protocols are:

AMBAPV

Models the AMBA® protocols AXI4, AXI3, AHB and APB.

AMBAPVACE

Models the AMBA® ACE and DVM protocols.

AMBAPVSignal

Models interrupts.

AMBAPVSignalState

Transfers and receives signal states.

AMBAPVValue

Models propagation of 32-bit integer values between components.

AMBAPVValue64

Models propagation of 64-bit integer values between components.

AMBAPVValueState

Permits a master to retrieve the current value from a slave, using 32-bit integer values.

AMBAPVValueState64

Permits a master to retrieve the current value from a slave, using 64-bit integer values.

There are ready-to-use components that provide you with conversions between protocols.

Related information

[AMBA-PV Extensions to TLM User Guide](#)

2.1.1 AMBAPV protocol

The AMBAPV protocol defines behaviors for single read and single write transactions. This covers Arm® AMBA® AXI5, AXI4, AXI3, AHB, and APB bus protocol families, at the PV level.

In addition, the AMBAPV protocol supports AMBA® protocol additional control information:

- Protection units.

- Exclusive access and locked access mechanisms.
- System-level caches.
- Atomic accesses, including exclusive accesses, locked accesses, and atomic transactions.

The LISA definition for the AMBAPV protocol is `$PVLIB_HOME/LISA/AMBAPVProtocol.lisa`. The behaviors of the protocol are:

read()

This optional slave behavior completes a single read transaction at the given address for the given size in bytes. Additional AMBA® protocol control information can be specified using the `ctrl` parameter. The `socket_id` parameter must be set to 0 in this context.

```
read(int socket_id, const sc_dt::uint64 &addr,
     unsigned char *data, unsigned int size,
     const amba_pv::amba_pv_control *ctrl,
     sc_core::sc_time &t):
    amba_pv::amba_pv_resp_t
```

write()

This optional slave behavior completes a single write transaction at the given address with specified data and write strobes. The size of the data is specified in bytes. Additional AMBA® protocol control information can be specified using the `ctrl` parameter. The `socket_id` parameter must be set to 0 in this context.

```
write(int socket_id, const sc_dt::uint64 &addr,
      unsigned char *data, unsigned int size,
      const amba_pv::amba_pv_control *ctrl,
      unsigned char *strb, sc_core::sc_time &t):
    amba_pv::amba_pv_resp_t
```

debug_read()

This optional slave behavior completes a debug read transaction from a given address without causing any side effects. Specify the number of bytes to read in the `length` parameter. The number of successfully read values is returned. Additional AMBA® protocol control information can be specified in the `ctrl` parameter. The `socket_id` parameter must be set to 0 in this context. This behavior is empty by default and returns 0.

```
debug_read(int socket_id, const sc_dt::uint64 &addr,
           unsigned char *data, unsigned int length,
           const amba_pv::amba_pv_control *ctrl):
    unsigned int
```

debug_write()

This optional slave behavior completes a debug write transaction to a given address without causing any side effects. Specify the number of bytes to write in the `length` parameter. The number of successfully written values is returned. Additional AMBA® protocol control information can be specified in the `ctrl` parameter. The `socket_id` parameter must be set to 0 in this context. This behavior is empty by default and returns 0.

```
debug_write(int socket_id, const sc_dt::uint64 &addr,
            unsigned char *data, unsigned int length,
```

```
const amba_pv::amba_pv_control *ctrl):
    unsigned int
```

b_transport()

This is an optional slave behavior for blocking transport. It completes a single transaction using the blocking transport interface. The `amba_pv::amba_pv_extension` must be added to the transaction before calling this behavior. The `socket_id` parameter must be set to 0 in this context.

```
b_transport(int socket_id, amba_pv::amba_pv_transaction &trans, sc_core::sctime
&t)
```

transport_dbg()

This optional slave behavior implements the TLM debug transport interface. An `amba_pv::amba_pv_extension` object must be added to the transaction before calling this behavior. The `socket_id` parameter must be set to 0 in this context.

```
transport_dbg(int socket_id, amba_pv::amba_pv_transaction &trans,
    sc_core::sctime &t):
    unsigned int
```

get_direct_mem_ptr()

This optional slave behavior requests a DMI access to a given address. It returns a reference to a DMI descriptor that contains the bounds of the granted DMI region. Returns `true` if a DMI region is granted, `false` otherwise.

The `amba_pv::amba_pv_extension` must be added to the transaction before calling this behavior. The `socket_id` parameter must be set to 0 in this context.

```
get_direct_mem_ptr(int socket_id, amba_pv::amba_pv_transaction &trans,
    tlm::tlm_dmi &dmi_data):
    bool
```

atomic_store() and atomic_load()

These optional slave behaviors complete an AtomicStore or AtomicLoad transaction with the specified data. The data is used by the atomic transaction in the specified endianness.

The only difference between the two behaviors is that AtomicStore does not return data, while AtomicLoad returns the original data.

```
atomic_store(int socket_id, const sc_dt::uint64 &addr,
    unsigned char * data, unsigned int length,
    unsigned int size, const amba_pv::amba_pv_control * ctrl,
    amba_pv::amba_pv_atomic_subop_t subop,
    amba_pv::amba_pv_atomic_endianness_t endianness, sc_core::sc_time &t) :
    amba_pv::amba_pv_resp_t

atomic_load(int socket_id, const sc_dt::uint64 &addr,
    unsigned char * data, unsigned int length, unsigned int size,
    const amba_pv::amba_pv_control * ctrl,
    amba_pv::amba_pv_atomic_subop_t subop,
    amba_pv::amba_pv_atomic_endianness_t endianness, sc_core::sc_time &t) :
    amba_pv::amba_pv_resp_t
```

atomic_swap()

This optional slave behavior completes an AtomicSwap transaction with the specified data, which is written to the specified address. The original data is returned.

```
atomic_swap(int socket_id, const sc_dt::uint64 & addr,
            unsigned char * data, unsigned int length, unsigned int size,
            const amba_pv::amba_pv_control * ctrl, sc_core::sc_time & t) :
            amba_pv::amba_pv_resp_t
```

atomic_compare()

This optional slave behavior completes an AtomicCompare transaction with the specified compare value and swap value. If the compare value equals the values at the given address, the swap value is written to the addressed location.

It always returns the original data at the target address.

```
atomic_compare(int socket_id, const sc_dt::uint64 & addr,
              unsigned char * data, unsigned int length, unsigned int size,
              const amba_pv::amba_pv_control * ctrl, sc_core::sc_time & t) :
              amba_pv::amba_pv_resp_t
```

invalidate_direct_mem_ptr()

This optional master behavior invalidates a DMI request. It invalidates DMI pointers that were previously established for the given DMI region. The `socket_id` parameter is 0 in this context.

```
invalidate_direct_mem_ptr(int socket_id, sc_dt::uint64 start_range,
                          sc_dt::uint64 end_range)
```

The generic payload data is formatted as an array of bytes in order of ascending bus address. This means that irrespective of the host machine endianness or modeled bus width:

- A little endian master must write the bytes of a word in increasing significance as the array index increases.
- A big endian master must write the bytes of a word in decreasing significance as the array index increases.

A master or slave whose endianness does not match the endianness of the host machine must endian swap any access to the payload data that is wider than one byte. The same byte ordering rule applies to memory accesses using DMI pointers.

2.1.2 AMBAPVACE protocol

This protocol defines behaviors for bus transactions. This covers Arm® AMBA® ACE and DVM bus protocol families, all at the PV level.

In addition, this protocol provides support for AMBA® protocol additional extension information:

- Secure and privileged accesses.

- Atomic accesses.
- System-level caching and buffering control.
- Cache coherency transactions (ACE-Lite).
- Bi-directional cache coherency transactions (ACE).
- Distributed virtual memory transactions (DVM).

The behaviors of the protocol are:

b_transport()

This slave behavior implements the TLM blocking transport interface. An `amba_pv::amba_pv_extension` object must be added to the transaction before calling this behavior. The `socket_id` parameter must be set to 0 in this context.

```
b_transport(int socket_id, amba_pv::amba_pv_transaction &trans, sc_core::sctime
&t)
```

transport_dbg()

This optional slave behavior implements the TLM debug transport interface. You must add an `amba_pv::amba_pv_extension` object to the transaction before calling this behavior, setting the `socket_id` parameter to 0 in this context.

```
transport_dbg(int socket_id, amba_pv::amba_pv_transaction &trans,
sc_core::sctime &t): unsigned int
```

get_direct_mem_ptr()

This optional slave behavior is for requesting a DMI access to a given address. It returns a reference to a DMI descriptor that contains the bounds of the granted DMI region. Returns true if a DMI region is granted, false otherwise. You must add an `amba_pv::amba_pv_extension` object to the transaction before calling this behavior, setting the `socket_id` parameter to 0 in this context.

```
get_direct_mem_ptr(int socket_id, amba_pv::amba_pv_transaction &trans,
tlm::tlm_dmi &dmi_data): bool
```

b_snoop()

This master behavior implements an upstream snooping TLM blocking transport interface. You must add an `amba_pv::amba_pv_extension` object to the transaction before calling this behavior, setting the `socket_id` parameter to 0 in this context.

```
b_snoop(int socket_id, amba_pv::amba_pv_transaction &trans, sc_core::sctime &t)
```

snoop_dbg()

This optional master behavior implements an upstream snooping TLM debug transport interface. You must add an `amba_pv::amba_pv_extension` object to the transaction before calling this behavior, setting the `socket_id` parameter to 0 in this context.

```
snoop_dbg(int socket_id, amba_pv::amba_pv_transaction &trans, sc_core::sctime
&t): unsigned int
```

invalidate_direct_mem_ptr()

Use this optional master behavior to invalidate a DMI request. It invalidates DMI pointers that were previously established for the given DMI region. The `socket_id` parameter is 0 in this context.

```
invalidate_direct_mem_ptr(int socket_id,  sc_dt::uint64 start_range,  
                           sc_dt::uint64 end_range)
```

The generic payload data is in the format of an array of bytes in order of ascending bus address. This means that irrespective of the host machine endianness or modeled bus width:

- A little endian master must write the bytes of a word in increasing significance as the array index increases.
- A big endian master must write the bytes of a word in decreasing significance as the array index increases.

A master or slave whose endianness does not match the endianness of the host machine must endian swap any access to the payload data that is wider than one byte. The same byte ordering rule applies to memory accesses using DMI pointers.

Special considerations for ACE and cache coherent interconnects

An ACE interconnect model must be able to cope with concurrent transactions in accordance with the hazard avoidance and prioritization rules in the ACE specification. Any external bus request, downstream transaction or upstream snoop transaction, can potentially cause a transaction to stall and the calling thread to be blocked, resulting in any number of other threads being scheduled.

To maintain memory coherency, apply these rules for debug transactions:

debug reads

The bus must return data that represents the values that the bus master expects to observe if it issues a bus read. This must not modify the state of any bus components.

debug writes

These must modify the contents of all copies of the location being accessed, so that a subsequent read from this location returns the data in the debug-write request. The debug write must not modify any other state, such as cache tags, clean/dirty/shared/unique MOESI state.

The implications for a coherent interconnect are that incoming debug transactions must be broadcast back upstream as debug snoop transactions to all ports other than the one the request came in on. Incoming debug snoops must propagate upwards. Debug reads can terminate as soon as they hit a cache. Debug writes must continue until they propagate to all possible copies of the location, including downstream to main memory.

For cases where a debug transaction hazards with non-debug transactions that are in-flight, the debug transaction must observe a weak memory-order model. Any component that can block a thread whilst responsible for the payload of an in-flight transaction must take particular care. In these cases, the debug transaction must be hazarded against the in-flight payload to ensure that debug reads do not return stale data and debug writes do not cause cache incoherency.

Only use DMI when you can guarantee that subsequent transactions do not result in any state transitions. This means, in general, do not use DMI for ACE coherent cacheable transactions.

2.1.3 AMBAPVSignal protocol

This protocol defines a single behavior to permit masters to change the state of signals such as interrupts. AMBA3 does not cover this behavior, but the AMBA-PV components do provide it.

set_state(int export_id, const bool &state): void

Transfers a signal state. The `export_id` parameter must be set to 0 in this context.

2.1.4 AMBAPVSignalState protocol

This protocol defines two behaviors that permit a master to change the state of signals such as interrupt and to retrieve the state of such signals from slaves. This behavior is not covered by AMBA3, but is provided with the AMBA-PV components.

set_state(int export_id, const bool &state): void

Transfers a signal state. The `export_id` parameter must be set to 0 in this context.

get_state(int export_id, tlm::tlm_tag<bool> * t): bool

Retrieves a signal state. The `export_id` parameter must be set to 0, and the `t` parameter must be set to `NULL`, in this context.

2.1.5 AMBAPVValue protocol

This protocol models propagation of 32-bit integer values between components. Strictly speaking this behavior is not covered by AMBA3, but is provided with the AMBA-PV components.

set_state(int export_id, const uint32_t &value): void

Transfers a value. The `export_id` parameter must be set to 0 in this context.

2.1.6 AMBAPVValue64 protocol

This protocol models propagation of 64-bit integer values between components. Strictly speaking this behavior is not covered by AMBA3, but is provided with the AMBA-PV components.

set_state(int export_id, const uint64_t &value): void

Transfers a value. The `export_id` parameter must be set to 0 in this context.

2.1.7 AMBAPVValueState protocol

This protocol permits propagation of 32-bit integer values between components and their retrieval from slaves. Strictly speaking this behavior is not covered by AMBA3, but is provided with the AMBA-PV components.

set_state(int export_id, const uint32_t &value):void

Transfers a value. The `export_id` parameter must be set to 0 in this context.

get_state(unsigned int export_id, tlm::tlm_tag<uint32_t> *t): uint32_t

Retrieves a value. The `export_id` parameter must be set to 0 and the `t` parameter must be set to `NULL`, in this context.

2.1.8 AMBAPVValueState64 protocol

This protocol permits propagation of 64-bit integer values between components and their retrieval from slaves.

Strictly speaking this behavior is not covered by AMBA3, but is provided with the AMBA-PV components.

set_state(int export_id, const uint64_t &value): void

Transfers a value. The `export_id` parameter must be set to 0 in this context.

get_state(int export_id, tlm::tlm_tag<uint64_t> *t): uint64_t

Retrieves a value. The `export_id` parameter must be set to 0, and the `t` parameter must be set to `NULL`, in this context.

2.2 Clocking protocols

The clocking components and protocols provide a mechanism for systems to regulate the execution rate of components.

All clocking components that communicate use an opaque `ClockSignal` protocol. `ClockSignal` protocols have no behaviors that the user can invoke.

Related information

[Clocking components](#) on page 160

2.2.1 ClockRateControl protocol

This protocol sets the ratio of this component.

- `set(uint32_t mul, uint32_t div) : void`
- `set64(uint64_t mul, uint64_t div) : void`

Set the multiplier and divider that determine the clock divider ratio:

```
clk_outfreq= clk_infreq * mul / div
```

2.2.2 TimerCallback protocol

This protocol invokes a behavior on the component that set the timer. Do not use the `timer_control` port of the timer during the invoked behavior.

signal(): uint32_t

Invoked by the timer when the timer countdown expires. The invoked behavior returns the number of ticks after which it will be called again or 0 to make the timer one-shot.

2.2.3 TimerCallback64 protocol

This protocol invokes a behavior on the component that set the timer. Do not use the `timer_control` port of the timer during the invoked behavior.

signal(): uint64_t

Invoked by the timer when the timer countdown expires. The invoked behavior returns the number of ticks after which it will be called again or 0 to make the timer one-shot.

2.2.4 TimerControl protocol

This protocol controls the actions of the component. It permits a timer to be set to schedule a callback after a given number of ticks at the rate of the clock input.

If a timer is set while it is counting, it starts counting the new number of ticks without sending the original callback. Canceling a timer when it is not active has no effect.

set(uint32_t ticks): void

Set the timer to count down the given number of ticks.

cancel()

Cancel an active timer, preventing the callback being invoked.

isSet() : bool

Check whether a timer is set to generate a callback.

remaining() : uint32_t

Return how many ticks remain before the callback is invoked.

2.2.5 TimerControl64 protocol

This protocol controls the actions of the component. It permits a timer to be set to schedule a callback after a given number of ticks at the rate of the clock input.

If a timer is set while it is counting, it starts counting the new number of ticks without sending the original callback. Canceling a timer when it is not active has no effect.

set(uint64_t ticks): void

Set the timer to count down the given number of ticks.

cancel()

Cancel an active timer, preventing the callback being invoked.

isSet() : bool

Check whether a timer is set to generate a callback.

remaining() : uint64_t

Return how many ticks remain before the callback is invoked.

2.3 Debug interface protocols

LISA components can expose aspects of their internal state so that they become visible and usable in the debugger. Some aspects are supported by the native LISA language, and some are supported by debug interface protocols.

Registers and other state variables

See the LISA+ resource REGISTER declaration, especially the `read_function` and `write_function` attributes.

Memories and other memory-like objects

See the LISA+ resource MEMORY declaration, especially the `read_function` and `write_function` attributes.

Disassembly

See `CADIProtocol` and `CADIDisassemblerProtocol`.

Instruction count

See `CADIProtocol`.

Setting, configuring, and clearing of breakpoints

For example, code breakpoints, register breakpoints, watchpoints. See `CADIProtocol`.

Single stepping and instruction stepping support

See `CADIProtocol`.

If displaying and editing memory and registers are sufficient, you need not implement `CADIProtocol` and `CADIDisassemblerProtocol`. The component must implement them to enable the other debug features.

CADIProtocol and CADIDisassemblerProtocol permit you to implement the features on a Component Architecture Debug Interface (CADI) level, where the component code takes responsibility for the implementation of these interfaces.

Related information

[Component Architecture Debug Interface User Guide](#)

[LISA+ Language for Fast Models Reference Guide](#)

2.3.1 CADIDisassemblerProtocol protocol

To support disassembly, implement all of these functions. None of them is optional.

These functions are in a different port, of type CADIDisassemblerProtocol, that can have any name and only need to be implemented when disassembly must be exposed in the debugger. The functionality of this port is then exposed by CADIProtocol::CADIGetDisassembler(). See CADIProtocol for information on how to use this port and CADIDisassemblerAdapter.

In the function slave behavior GetType(): eslapi::CADIDisassemblerType; components must always return eslapi::CADI_DISASSEMBLER_TYPE_STANDARD.

The function slave behavior GetModeCount(): uint32_t; returns the number of supported disassembler modes, and at least 1 mode must be returned.

The function slave behavior GetModeNames(eslapi::CADIDisassemblerCB *callback_); returns information about all supported modes. A component that only supports one mode calls, for example, callback_>ReceiveModeName(0, "Normal"); only once. This is similar for multiple modes with different names and ids.

The function slave behavior GetCurrentMode(): uint32_t; returns the most suitable mode of disassembly at the time, based on the current state variables of the component:

For the function:

```
slave behavior GetSourceReferenceForAddress(eslapi::CADIDisassemblerCB *callback_,
const eslapi::CADIAddr_t &address): eslapi::CADIDisassemblerStatus;
```

components must return eslapi::CADI_DISASSEMBLER_STATUS_ERROR;

For the function:

```
slave behavior GetAddressForSourceReference(const char *sourceFile, uint32_t
sourceLine, eslapi::CADIAddr_t &address): eslapi::CADIDisassemblerStatus;
```

components must return eslapi::CADI_DISASSEMBLER_STATUS_ERROR;

The following function is the main disassembler function:

```
slave behavior GetDisassembly(eslapi::CADIDisassemblerCB *callback_,
                             const eslapi::CADIAddr_t &address,
                             eslapi::CADIAddr_t &nextAddr,
                             const uint32_t mode,
                             uint32_t desiredCount):
    eslapi::CADIDisassemblerStatus;
```

The component must call `callback_` for all disassembler lines for the specified address and `desiredCount`, and it must finally set `nextAddr` to the next disassembled address at that point after the requested block.

```
// Query if an instruction is a call instruction
slave behavior GetInstructionType(const eslapi::CADIAddr_t
&address, eslapi::CADIDisassemblerInstructionType &insn_type):
    eslapi::CADIDisassemblerStatus;
```

Components must return `insn_type = eslapi::CADI_DISASSEMBLER_INSTRUCTION_TYPE_NOCALL;` and return `eslapi::CADI_DISASSEMBLER_STATUS_OK;`

Related information

[CADIProtocol protocol](#) on page 76

[Component Architecture Debug Interface User Guide](#)

2.3.2 CADIProtocol protocol

This protocol supports debugging. To add breakpoint support, implement `CADIBpt...()` functions.



Note

All CADIProtocol protocol behaviors, that is, all sets of functionalities, are optional. A component only has to implement the set of functions for the functionality that it intends to support.

You must define an internal slave port of this type, and the name of this port must always be `cadi_port`. In this port, you can implement this functionality:

```
optional slave behavior CADIBptGetList(uint32_t, uint32_t, uint32_t *,
    eslapi::CADIBptDescription_t *):eslapi::CADIReturn_t;
optional slave behavior CADIBptRead(eslapi::CADIBptNumber_t,
    eslapi::CADIBptRequest_t *):eslapi::CADIReturn_t;
optional slave behavior CADIBptSet(eslapi::CADIBptRequest_t *,
    eslapi::CADIBptNumber_t *):eslapi::CADIReturn_t;
optional slave behavior CADIBptClear(eslapi::CADIBptNumber_t):eslapi::CADIReturn_t;
optional slave behavior CADIBptConfigure(eslapi::CADIBptNumber_t,
    eslapi::CADIBptConfigure_t):eslapi::CADIReturn_t;
optional slave behavior CADIModifyTargetFeatures(eslapi::CADITargetFeatures_t
    *):eslapi::CADIReturn_t;
```

You need to implement all of these functions to enable any kind of breakpoint for the component. The component needs to maintain and keep track of all existing breakpoints. For example:

- `CADIBptSet()` and `CADIBptClear()` add and remove breakpoints.
- `CADIBptConfigure()` enable and disable breakpoints.
- `CADIBptGetList()` and `CADIBptRead()` return the current state of the breakpoint list.

The component must also implement `CADIModifyTargetFeatures`. This function permits you to override the automatic default `CADITargetFeatures_t` that System Generator provides for this component just before it is returned to the debugger. Specifically, a component that wants to support any kind of breakpoint must override the `handledBreakpoints` and `nrBreakpointsAvailable` fields of `CADITargetFeatures_t`. For example:

```
targetFeatures->handledBreakpoints = CADI_TARGET_FEATURE_BPT_PROGRAM |
    CADI_TARGET_FEATURE_BPT_REGISTER;
    // code and register breakpoints supported
targetFeatures->nrBreakpointsAvailable = 0x7fffffff;
    // virtually infinite number of breakpoints supported.
```

By default, LISA components do not support any type of breakpoints. By implementing `CADIBpt...` functions, you can add breakpoint support. In addition to implementing the stated functions, the component must call `simBreakpointHit(bptNumber)` and then `simHalt()` when an enabled breakpoint is hit. On a breakpoint hit the component must first call `simBreakpointHit()` for each breakpoint that was hit (one or more, usually just one) and then call `simHalt()` once after all `simBreakpointHit()` calls. The `simHalt()` call must always be the last call in the sequence.

A component that wants to provide disassembly must implement the following `CADIGetDisassembler()` behavior and return a `CADIDisassembler` interface implementation. This automatically follows behind the `CADI::CADIGetDisassembler()` and the `CADI::ObtainInterface("eslapi.CADIDisassembler2")` functions.

```
optional slave behavior CADIGetDisassembler():eslapi::CADIDisassembler*;
```

To do this, instantiate a `CADIDisassemblerAdapter` object in behavior `init` and return its address in this `CADIGetDisassembler()` function. This object must point to an internal slave port that implements the `CADIDisassemblerProtocol` protocol.

Skeleton code for implementing disassembly:

```
component FOO
{
    behavior init()
    {
        disassemblerAdapter = new
        CADIDisassemblerAdapter(disassPort.getAbstractInterface());
        // ...
    }
    internal slave port <CADIProtocol> cadi_port
    {
        slave behavior CADIGetDisassembler():eslapi::CADIDisassembler*
        {
            return disassemblerAdapter;
        }
        // ...
    }
    internal slave port<CADIDisassemblerProtocol> disassPort
```

```
{
    // ...
}
```

The following function implements the instruction stepping a component. It must set up an internal state that stops the simulation when the requested number of instructions is executed completely (exactly like a breakpoint). It must call `simRun()` from within `CADIExecSingleStep()` after setting up this stepping state, and later it must call `simHalt()` when the execution of the required number of instructions finishes.

```
optional slave behavior CADIExecSingleStep(uint32_t instructionCount, int8_t
stepCycle, int8_t stepOver):eslapi::CADIReturn_t;
```

The following function is for debugging purposes only. Do not implement it. The function must not alter the state of any component in any way.

```
optional slave behavior callbackModeChange(uint32_t newMode, eslapi::CADIBptNumber_t
bptNumber);
```

By implementing any of the following functions, the component can enable the instruction and cycle count display:

```
optional slave behavior CADIGetInstructionCount(uint64_t
&instructionCount):eslapi::CADIReturn_t;
optional slave behavior CADIGetCycleCount(uint64_t &instructionCount, bool
systemCycles):eslapi::CADIReturn_t;
```



Fast Models systems are not cycle accurate, so you usually only implement an instruction counter, if at all.

2.4 Peripheral protocols

This section describes the peripheral protocols.

2.4.1 AudioControl protocol

This protocol has get and release audio buffer behaviors.

getPVAudioBuffer

Get an underlying host buffer for audio output.

releasePVAudioBuffer

Release an underlying host buffer.

2.4.2 CharacterLCD protocol

This protocol has the behaviors `setLayoutInfo` and `draw`.

setLayoutInfo

Sets the width and height of the touchscreen.

draw

Sets the character information.

2.4.3 FlashLoaderPort protocol

This protocol initializes the flash contents at model startup and saves flash contents to a file when the model terminates.

loadFlashFile(FlashLoader *) : uint32

Initiate loading of the flash contents.

saveFlashFile(FlashLoader *) : uint32

Save the flash contents to a file.

2.4.4 GUIPollCallback protocol

This protocol defines one method that signals to the visualization component the end of the update period. You can invoke this callback even when the simulation is stopped.

gui_callback(): void

This is sent by the GUIPoll component, at the configured period.

2.4.5 ICS307Configuration protocol

This protocol sets the divider ratio of an ICS307 component at runtime. The output clock rate alters accordingly and any dependent components react to the clock rate change according to their defined behavior.

setConfiguration(uint32_t vdw, uint32_t rdw, uint32_t od): void

Set the parameters for deriving the clock divider ratio.

vdw

Range: 0-255.

rdw

Range: 0-255.

od

Range: 0-7.

2.4.6 KeyboardStatus protocol

This protocol passes keyboard events to a component such as the PS2Keyboard component.

Events are only sent when the visualization window is in focus. Keyboard combinations that are filtered by the host OS such as **Ctrl+Alt+Del** are not detected by the visualization. See `$PVLIB_HOME/include/components/KeyCode.h` for a list of `ATKeyCode` values.

The protocol behaviors are:

keyDown(ATKeyCode code) : void

Sent when a key on the host keyboard is pressed.

keyUp(ATKeyCode code) : void

Sent when a key on the host keyboard is released.

Related information

[VEVisualisation ports](#) on page 4270

2.4.7 LCD protocol

This Visualisation Library signaling protocol provides the interface between an LCD controller peripheral, for example the PL110, and a visualization component. This permits the LCD controller to render the framebuffer contents into a region of the visualization GUI.

LISA visualization components can provide any number of LCD ports. The implementations of these behaviors can delegate the calls to appropriate methods on the `VisRenderRegion` class.

The behaviors are:

lock() : VisRasterLayout *

Lock the raster region of the LCD in preparation for rendering.

unlock()

Unlock the raster region, ready to be updated on the screen.

update(int x, int y, unsigned int w, unsigned int h)

Update the selected rectangular area on screen from the raster buffer.

setPreferredLayout(unsigned int width, unsigned int height, unsigned int depth)

A request from the LCD controller to set the preferred size for the raster region, to match the timing parameters used by the LCD controller.

Related information

[VEVisualisation ports](#) on page 4270

2.4.8 LCDLayoutInfo protocol

This protocol has the behavior `setLayoutInfo`.

setLayoutInfo

Sets the width and height of the touchscreen.

2.4.9 MMC_Protocol protocol

This protocol describes an abstract, untimed interface between an MMC controller and an MMC or SD card.

The protocol contains methods that must be implemented by the master (controller) and some that must be implemented by the slave (card). This protocol is used by the reference PL180 MCI and MMC models. For further information on the protocol implementation, see the source file, `$PVLIB_HOME/LISA/MMC_Protocol.lisa`.

Use of this protocol assumes knowledge of the MultiMediaCard specification, available from the MultiMediaCard Association, www.mmca.org.

The protocol has these behaviors:

cmd

Commands are sent from the controller to the card using this behavior, which is implemented by the card model. The MMC command is sent with an optional argument. The card responds as defined by the MMC specification. The controller model checks that the response type matches expectations, and updates its state appropriately. The transaction-level protocol does not model start/stop bits or CRCs on the command/response payload.

For data transfer in the card to controller direction:

Rx

After the host and controller have initiated a read through the command interface, the card calls the `Rx` behavior on the controller to provide the read data. The call provides a pointer and a length. The ARM MMC reference model simulates device read latency by waiting a number of clock cycles prior to calling this behavior. If the controller is unable to accept the data, or wants to force a protocol error, it can return false in response to this behavior.

Rx_rdy

A handshake, used by the controller to inform the card that the controller is ready to receive more data. The Arm MMC reference model does not time out, so waits indefinitely for this handshake in a multiple block data transfer.

For data transfer in the controller to card direction:

Tx

After the host and controller have initiated a write through the command interface, the card calls the `Tx` behavior on the controller. The call provides a pointer to an empty buffer to be

written, and a length. The ARM MMC reference model simulates device write latency by waiting a number of clock cycles prior to each buffer being offered.

Tx_done

The controller calls this behavior on the card when the block has been written. The card model can then commit the data to its persistent storage.

The card model must also implement:

cmd_name

This behavior returns the name of the command issued. A card must implement this behavior, but is free to return an empty string for all requests. Only call this behavior for diagnostic messages.

2.4.10 MouseStatus protocol

This protocol passes mouse movement and button events to another component such as the PS2Mouse component.

Events are only sent when the visualization window is in focus.

The protocol behaviors are:

mouseMove(int dx, int dy) : void

This is sent when the host mouse is moved. Mouse movement events are always relative.

mouseButton(uint8_t button, bool down) : void

This is sent when a button on the host mouse is pressed or released.

`button` indicates which button has been pressed or released and is typically 0, 1, or 2 but can be anything up to 7 depending on the OS and attached mouse.

`down` is true if a button is pressed and false if released.

Related information

[VEVisualisation ports](#) on page 4270

2.4.11 PL080_DMAC_DmaPortProtocol protocol

This protocol provides methods to permit handshaking between peripherals and the DMA controller.

request(uint32 request) : void

Passes requests from a peripheral to the DMA controller. The request is a bitfield with the low four bits defined. The request is level-sensitive and latched internally by the DMA controller. It is sampled and interpreted in a manner dependent on the target channel and configured flow control.

0: PL080_REQ_BURST

Burst transfer request.

1: PL080_REQ_SINGLE

Single transfer request.

2: PL080_REQ_LBURST

Last burst request.

3: PL080_REQ_LSINGLE

Last single request.

response(uint32 response): void

Passes responses from the DMA controller to peripherals. The response is a bitfield with the low two bits defined. The response is transient rather than level-sensitive.

0: PL080_RES_TC

Terminal count response.

1: PL080_RES_CLR

Clear request response.

2.4.12 PS2Data protocol

This protocol is for communication between the KMI and a PS/2-like device.

For efficiency, the interface is a parallel byte interface rather than a serial clock/data interface. The behaviors are:

setClockData(enum ps2clockdata) : void

Used by the KMI to simulate forcing the state of the data/clock lines, to indicate whether it is able to receive data, wants to send a command, or is inhibiting communication.

getData() : uint8

Used by the PS/2 device to get command data from the KMI.

putData(uint8 data): void

Used by the PS/2 device to send device data to the KMI.

2.4.13 PVBusSlaveControl protocol

The PVBusSlaveControl protocol enables you to access and modify the underlying memory that PVBusSlave controls.

- `setFillPattern(uint32_t fill1, uint32_t fill2)`

This sets a two-word alternating fill pattern to be used by uninitialized memory.

- `setAccess(pv::bus_addr_t base, pv::bus_addr_t top, pv::accessType type, pv::accessMode mode)`

This reconfigures handling for a region of memory.

`base` (inclusive value) and `top` (exclusive value) specify the address range to configure.

`type` selects what types of bus access must be reconfigured. It can be one of:

- `pv::ACCESSTYPE_READ`
- `pv::ACCESSTYPE_WRITE`
- `pv::ACCESSTYPE_RW`

`mode` controls what happens when an address is accessed. Legal values for the `pv::accessMode` enumeration are:

`pv::ACCESSMODE_MEMORY`

Act as memory. The PVBUSlave manages the underlying storage to provide memory for the selected address range, which can be ROM or RAM, depending on how you configure it to handle bus write transactions.

`pv::ACCESSMODE_DEVICE`

Act as a device. Requests to the selected address range are routed to the PVBUSlave device port, where the necessary behavior can be implemented by the component.

`pv::ACCESSMODE_ABORT`

Generate bus abort signals for any accesses to the selected address range.

`pv::ACCESSMODE_IGNORE`

Ignore accesses to the selected address range. Bus read requests return 0.

- `getReadStorage(pv::bus_addr_t address, pv::bus_addr_t *limit) : const uint8_t *`
- `getWriteStorage(pv::bus_addr_t address, pv::bus_addr_t *limit) : uint8_t *`

These two methods permit you to access the underlying storage that PVBUSlave allocates to implement a region of memory.

The return value is a pointer to the byte that represents the storage corresponding to the address of `base`.

The `limit` pointer returns the device address for the limit of the accessible memory.

The pointer value returned is not guaranteed to remain valid indefinitely. Bus activities, or other calls to the control port, might invalidate the pointer. For example, if a burst transaction straddles a 4KB boundary, PVBUSlave might reallocate the 4KB regions to be contiguous.

- `provideReadStorage(pv::bus_addr_t device_base, pv::bus_addr_t device_limit, const uint8_t *storage)`
- `provideWriteStorage(pv::bus_addr_t device_base, pv::bus_addr_t device_limit, uint8_t *storage)`
- `provideReadWriteStorage(pv::bus_addr_t device_base, pv::bus_addr_t device_limit, uint8_t *storage)`

These methods enable you to allocate blocks of memory that the PVBUSlave can use to manage regions of RAM/ROM. Only use these methods when you require a high degree of control over memory, such as when you require a device to map specific regions of host memory into the simulation.

The memory region pointed to by `storage` must be large enough to contain `(limit - base)` bytes.

After these calls, PVBUSlave controls access to the underlying memory. The owner must call `getWriteStorage()` before modifying the memory contents and `getReadStorage()` before reading the memory contents.

- `provideReadStorageEx(pv::bus_addr_t device_base, pv::bus_addr_t device_limit, const uint8_t *storage, double latency)`
- `provideWriteStorageEx(pv::bus_addr_t device_base, pv::bus_addr_t device_limit, const uint8_t *storage, double latency)`
- `provideReadWriteStorageEx(pv::bus_addr_t device_base, pv::bus_addr_t device_limit, uint8_t *storage, double read_latency, double write_latency)`

These methods take additional parameters to specify average latencies (in seconds) per byte, used when Timing Annotation is enabled.

In all other aspects they behave the same as `provideReadStorage()`, `provideWriteStorage()` and `provideReadWriteStorage()`, respectively.

- `slave behavior getRegionIterHandle(): uint32_t;`
- `slave behavior getNextRegionInfo(uint32_t iter_handle, pv::PVBUSlaveRegionInfo *info) : bool;`
- `slave behavior closeRegionIterHandle(uint32_t iter_handle);`

These methods form an iterator-like API that allows a PVBUSlave providing storage to report all the regions of the address space that have backing store.

The iteration begins by calling `getRegionIterHandle()`. This allocates an iterator and if successful returns a nonzero `iter_handle` to identify it.

The caller can then repeatedly call `getNextRegionInfo()` with `iter_handle`. If it finds a region, the behavior returns `true` and writes to the `info` struct if the pointer is non-null. Access the data in the region using `getReadStorage()` OR `getWriteStorage()`.

The implementation can return regions in any order. They can be of any size or alignment, but must not overlap.

The implementation need not report allocated regions that are filled entirely with the default fill pattern, or allocated regions that contain only the data they had at simulation start.

On reaching the last region, the iterator closes automatically. If the handle is invalid or there are no further regions, the behavior returns `false`.

A caller can close an iterator opened by `getRegionIterHandle()` at any time using `closeRegionIterHandle()`. This deallocates the iterator, and further uses of the handle are invalid.

2.4.14 PVDevice protocol

The PVDevice protocol enables you to implement support for memory-mapped device registers. Call the two methods through the device port on the PVBUSlave to handle bus read/write transactions.

read(pv::ReadTransaction): pv::Tx_Result

This method permits a device to handle a bus read transaction.

write(pv::WriteTransaction): pv::Tx_Result

This method permits a device to handle a bus write transaction.

The PVDevice protocol uses two behaviors to differentiate between transactions originating from the processor (loads and stores) and transactions originating from an attached debugger:

slave behavior debugRead(pv::ReadTransactiontx) : pv::Tx_Result

This method enables the device to handle a debug read transaction.

slave behavior debugWrite(pv::WriteTransactiontx) : pv::Tx_Result

This method enables the device to handle a debug write transaction.

The `debugRead` and `debugWrite` behaviors are called for all debug transactions.

For an example component that uses this protocol see `$PVLIB_HOME/examples/LISA/BusComponents/BitLatch.lisa`.

Related information

[PVBUS C++ transaction and Tx_Result classes](#) on page 56

2.4.15 PVTransactionMaster protocol

This protocol instantiates a `pv::TransactionGenerator` object from a PVBUSMaster.

createTransactionGenerator() : pv::TransactionGenerator *

This behavior instantiates a new transaction generator to control the bus master. A caller can allocate as many TransactionGenerators as it wants. It is up to the caller to delete TransactionGenerators when they are no longer required. For example:

```
behavior init() {
    tg = master.createTransactionGenerator();
}

behavior terminate() {
    delete tg;
}
```

```
}

```

2.4.16 SerialData protocol

This protocol is implemented as a parallel interface for efficiency. All communication is driven by the master port.

This protocol has behaviors:

dataTransmit(uint16_t data) : void

Used by the master to send data to the slave.

Table 2-1: Bits for dataTransmit()

Bits	Function
15:8	Reserved
7:0	Transmit data

dataReceive(void) : uint16_t

Used by the master to receive data from the slave.

Table 2-2: Bits for dataReceive()

Bits	Function
15:13	Reserved
12	Set when no data available for reading
11	Reserved
10	Break error
9:8	Reserved
7:0	Receive data

signalsSet(uint8_t signal) : void

Used by the master to get the current signal status.

Table 2-3: Bits for signalsSet()

Bits	Function
7	Out1
6	Out2
5	RTS
4	DTR
3:0	Reserved

signalsGet() : uint8_t

Used by the master to get the current signal status.

Table 2-4: Bits for signalsGet()

Bits	Function
7:4	Reserved
3	DCD
2	DSR
1	CTS
0	RI

2.4.17 SMMUv3AEMIdentifyProtocol protocol

This protocol is used by the SMMUv3AEM model to identify the SSD, StreamID, and SubStreamID of an incoming transaction. It is only used if the parameter `howto_identify` is set to "use-identify".

```
identify(unsigned tbu_number_, const pv::TransactionAttributes* attributes_, bool*
out_ssd_ns_, unsigned* out_streamid_, unsigned* out_substreamid_) : void`
```

Architecturally, a transaction comes into the SMMUv3AEM model with the following side band signals:

- Security State Determination (SSD):
 - 0**
Transaction belongs to a device controlled by the secure world.
 - 1**
Transaction belongs to a device controlled by the non-secure world.
- StreamID.
- SubStreamID and SubStreamID valid. If you set `*out_substreamid_ = ~0u`, that is interpreted as no SubStreamID because SubStreamIDs are 20 or fewer bits.

How these are transported in the system is SoC-dependent.

The SMMUv3AEM requires that the SoC provides a way of determining this information by implementing the `identify()` function.

2.4.18 TZFilterControl protocol

This protocol controls the communication between filter units and control registers in the APB control block.

```
checkPermission(const pv::TransactionAttributes *attributes_, pv::bus_addr_t
page_base_, bool is_read_, pv::RemapRequest &req_, bool &abort_on_error_) : bool
```

Check the permission of the transactions filtered by the filter unit. Optional slave behavior.

isEnabled() : bool

Check if the filter unit is enabled or not. The APB control block controls the unit. Slave behavior.

isSecureSlave() : bool

Check if the connected slave is secure or not. Optional slave behavior.

setConfig(bool rd_spec_enable, bool wr_spec_enable, uint32_t action) : void

Pass the configurations to the filter. Optional master behavior.

2.4.19 VirtualEthernet protocol

This protocol has the `sendToSlave` and `sendToMaster` behaviors.

sendToSlave(EthernetFrame *frame)

Send an Ethernet frame to the slave port.

sendToMaster(EthernetFrame *frame)

Send an Ethernet frame to the master port.

The Ethernet frame class encapsulates an Ethernet frame in a broken-up format that is more accessible by components. For information on the class definition, see the `EthernetFrame.h` header file located in `$PVLIB_HOME/include/components/VirtualEthernet/Protocol`.

2.5 Power management protocols

This section describes the power management protocols.

2.5.1 PChannel protocol

Communicates power state changes between a power controller and a device.

The behaviors of the protocol are:

pactive (uint32_t pstate) : void

This master behavior is implemented by a power controller. A device calls this method to give a hint to the power controller that it can change to a particular power state. A power controller can then take appropriate action, typically communicating with the device by calling `device.prequest(new_power_state)`.

The power state is type `uint32_t` because it is the responsibility of the system using PChannels to enumerate the power states that it supports. For example, Armv8-A cores use the following enumeration for power states:

```
enum { OFF = 0, OFF_EMU, MEM_RET, MEM_RET_EMU, LOGIC_RET, FULL_RET, MEM_OFF,
      FUNC_RET, ON, WARM_RST, DBG_RECOV }
```

prequest (uint32_t pstate) : sg::PChannel::presp_t

This slave behavior is implemented by a device, for instance a core. A power controller typically calls this method and checks for the response from the device, which can either be ACCEPT OR DENY.

sg::PChannel::presp_t

This enumeration provides two values, ACCEPT and DENY. It is returned by the `prequest()` method, depending on the state requested and the current state of the core.

Usage

You can use PChannels to replace STANDBYWFI and STANDBYWFE signaling.

For example, using STANDBYWFI OR STANDBYWFE:

- Core drives STANDBYWFI signal HIGH.
- Power controller performs logic <x>.

Equivalent behavior using PChannels:

- Core calls `pactive(OFF)`.
- Power controller calls `prequest(OFF)` to change the core to OFF.
- Power controller performs logic <x>.
- To wake up the core, the power controller calls `prequest(ON)`.

Examples

- For a LISA+ example that uses PChannel, see `$PVLIB_HOME/examples/LISA/VP_PChannel/`.
- For a SystemC example that uses PChannel, see `$PVLIB_HOME/examples/SystemCExport/EVS_Components/EVS_PChannel/`.

2.6 Processor protocols

This section describes the processor protocols.

2.6.1 CoprocBusProtocol protocol

This protocol connects a coprocessor implementation with a CPU component, for instance ARMCortexM33CT.

A coprocessor must derive from the coprocessor callback interface, `coprocessor`. It can implement the CDP, MCR, MRC, STC, LDC, MCRR, and MRRC instructions.

A coprocessor must be registered with a specific coprocessor number, by calling the `addCoproprocessor()` method. You can only register an external coprocessor that is not already present in the CPU. If no coprocessor has been registered with the coprocessor number encoded in an instruction, the CPU raises a NOCP fault.

To register coprocessor instruction implementations with the CPU, you must initialize the function pointers. For example, the following code passes the function pointers to the `Coprocessor` constructor. This code was taken from the `$PVLIB_HOME/examples/LISA/FVP_Coproc_Demo` example.

Registering a coprocessor

```
...
class TestValCoprocessor : public Coprocessor
{
public:
    protocol_CoprocBusProtocol * coproc_bus;
    uint32_t coproc_number;
    uint32_t cp_reg[2][NUM_CP_REG] = {{0}}; // [0][NUM_CP_REG] --> Secure, [1]
[ NUM_CP_REG ] --> Non-Secure
    TestValCoprocessor()
        : Coprocessor(this, test_CDP, nullptr, test_MCR, nullptr, test_MRC, nullptr,
test_LDC, nullptr, test_STC, nullptr, test_MCRR, nullptr, test_MRRC, nullptr)
        , coproc_bus(nullptr)
        , coproc_number(0)
        {
    ...
};

PARAMETER { description("coprocessors number"), type(uint32_t), default(0x2),
min(0x0), max(16) } coprocessor_number; // CP num
TestValCoprocessor test_cp;
}

behaviour init
{
...
    if (coproc_bus.addCoprocessor.implemented())
    {
        coproc_bus.addCoprocessor(&test_cp, coprocessor_number);
    }
}
```

2.6.1.1 Coprocessor behaviors

A coprocessor can call the following master behaviors:

addCoprocessor(Coprocessor*, int num) : void

Registers the coprocessor with the CPU. `num` identifies which coprocessor to register it as.

removeCoprocessor(Coprocessor*, int num) : void

Unregisters the coprocessor from the CPU.

accessIsPriv(void) : bool

Checks whether the CPU state is privileged (true) or unprivileged (false).

accessIsNonSecure(void) : bool

Checks the security state of the CPU, either true for non-secure, or false for secure.

2.6.1.2 Coprocessor callback functions

A coprocessor can implement callback functions with these signatures.

Each function returns a `CoprocState` value to indicate the new transaction state of the coprocessor, as listed in [CoprocState values](#).

2.6.1.2.1 CDP()

Perform a coprocessor data processing operation.

Prototype

```
CoprocState CDP(void* context, uint32_t inst)
```

Parameters

context

context that was registered with the coprocessor interface.

inst

the coprocessor instruction being executed.

Returns

A `CoprocState` value to indicate the new transaction state of the coprocessor, as listed in [CoprocState values](#).

2.6.1.2.2 MCR()

Perform a move to coprocessor register operation.

Prototype

```
CoprocState MCR(void* context, uint32_t inst, uint32_t data)
```

Parameters

context

context that was registered with the coprocessor interface.

inst

the coprocessor instruction being executed.

data

register contents.

Returns

A `CoprocState` value to indicate the new transaction state of the coprocessor, as listed in [CoprocState values](#).

2.6.1.2.3 MRC()

Perform a move from coprocessor register operation.

Prototype

```
CoprocState MRC(void* context, uint32_t inst, uint32_t* data)
```

Parameters

context

context that was registered with the coprocessor interface.

inst

the coprocessor instruction being executed.

data

pointer to word to fill with coprocessor register contents.

Returns

A `CoprocState` value to indicate the new transaction state of the coprocessor, as listed in [CoprocState values](#).

2.6.1.2.4 LDC()

Perform a load coprocessor register from memory operation.

Prototype

```
CoprocState LDC(void* context, uint32_t inst, uint32_t data, CoprocState state)
```

Parameters

context

context that was registered with the coprocessor interface.

inst

the coprocessor instruction being executed.

data

contents of current memory location to load into register.

state

current state in a sequence of transactions.

Returns

A `CoprocState` value to indicate the new transaction state of the coprocessor, as listed in [CoprocState values](#).

2.6.1.2.5 STC()

Perform a store coprocessor register to memory operation.

Prototype

```
CoprocState STC(void* context, uint32_t inst, uint32_t data, CoprocState state)
```

Parameters

context

context that was registered with the coprocessor interface.

inst

the coprocessor instruction being executed.

data

pointer to word to fill with coprocessor register contents to be transferred to memory.

state

current state in a sequence of transactions.

Returns

A `CoprocState` value to indicate the new transaction state of the coprocessor, as listed in [CoprocState values](#).

2.6.1.2.6 MCRR()

Perform a move to two coprocessor registers operation.

Prototype

```
CoprocState MCRR(void* context, uint32_t inst, uint32_t data1, uint32_t data2)
```

Parameters

context

context that was registered with the coprocessor interface.

inst

the coprocessor instruction being executed.

data1

first data word to load to a coprocessor register.

data2

second data word to load to a coprocessor register.

Returns

A `CoprocState` value to indicate the new transaction state of the coprocessor, as listed in [CoprocState values](#).

2.6.1.2.7 MRRC()

Perform a move from two coprocessor registers operation.

Prototype

```
CoprocState MRRC(void* context, uint32_t inst, uint32_t* data1, uint32_t* data2)
```

Parameters

context

context that was registered with the coprocessor interface.

inst

the coprocessor instruction being executed.

data1

pointer to first word to fill with coprocessor register contents.

data2

pointer to second word to fill with coprocessor register contents.

Returns

A `CoprocState` value to indicate the new transaction state of the coprocessor, as listed in [CoprocState values](#).

2.6.1.3 CoprocState values

A `CoprocState` enum value is returned by coprocessor callback functions to indicate the new transaction state of the coprocessor. It is also used as a parameter for `LDC` and `STC` callback functions.

Table 2-5: CoprocState values

Value	State label	Description
0	<code>CoprocOk</code>	Complete/Ok.
1	<code>CoprocUndef</code>	Undefined operation.
2	<code>CoprocAbort</code>	Data abort.
4	<code>CoprocFirst</code>	A parameter value for <code>LDC</code> and <code>STC</code> callback functions to indicate that this is the first data transfer in a sequence.
5	<code>CoprocNext</code>	A parameter value for <code>LDC</code> and <code>STC</code> callback functions to indicate that this is a subsequent data transfer in a sequence.
12	<code>CoprocNop</code>	Treat as a NOP .

2.6.2 CounterInterface protocol

This protocol connects the `cntvalueb` port on Generic Timer components to MemoryMappedCounterModule components.

This semi-opaque protocol is exportable across a SystemC interface using a custom bridge.

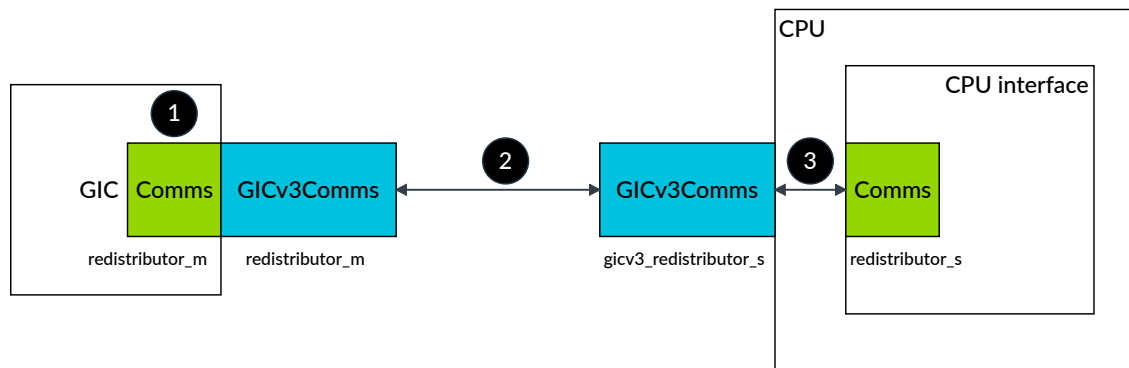
It is a LISA+ protocol.

2.6.3 GICv3Comms protocol

GICv3Comms is a protocol for communication between redistributors in the GIC and the CPU interface. It is supported by the GICv3 and GICv4 models. It is defined in `$PVLIB_HOME/LISA/GICv3Comms.lisa`.

Each GICv3Comms port in the LISA+ file is connected internally to a Comms port, which is a bi-directional communication port. Comms is derived from `sg::port`. The Comms port in the GIC is called `redistributor_m` and the one in the CPU interface is called `redistributor_s`, as shown in the following diagram:

Figure 2-1: GICv3Comms block diagram



Key:

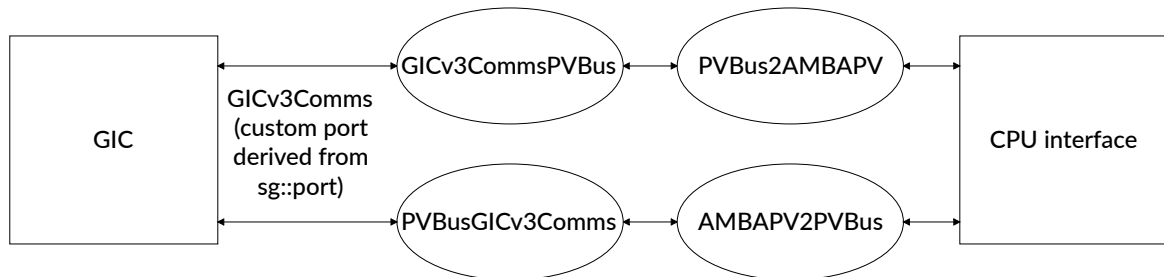
1. GIC contains a Comms port, `redistributor_m`, which the LISA+ file exposes as a GICv3Comms port, `redistributor_m`.
2. `redistributor_m` is connected to the GICv3Comms port in the CPU, `gicv3_redistributor_s`.
3. `gicv3_redistributor_s` forwards the connection to the CPU interface Comms port, `redistributor_s`.

Each redistributor in the GIC communicates through `redistributor_m` to each CPU interface. The number of redistributors in the GIC equals the number of PEs connected.

GIC commands, for example Deactivate, Activate, ActivateAcknowledge, GenerateSGI, Set or Clear interrupt, are communicated through Comms.

The PVBUSGICv3Comms and GICv3CommsPVBUS components convert between the GICv3Comms and PVBUS protocols. The following diagram shows how they can be used in distributed systems:

Figure 2-2: Conversion between GICv3Comms and PVBUS protocols



2.6.4 InstructionCount protocol

This protocol has the behaviors `getValue()` and `getRunState()`.

getValue() : uint64_t

Obtain the number of instructions executed by the processor.

getRunState() : uint32_t

Obtain the power/run status of the processor.

Table 2-6: Run state values

Value	State label	Description
0	UNKNOWN	Run status unknown, that is, simulation has not started
1	RUNNING	Processor running, is not idle and is executing instructions
2	HALTED	External halt signal asserted
3	STANDBY_WFE	Last instruction executed was WFE and standby mode has been entered
4	STANDBY_WFI	Last instruction executed was WFI and standby mode has been entered
5	IN_RESET	External reset signal asserted
6	DORMANT	Partial processor power down
7	SHUTDOWN	Complete processor power down

2.6.5 v8EmbeddedCrossTrigger_controlprotocol protocol

This protocol connects the Cross Trigger Interface (CTI) in processor components to platform-level Cross Trigger Matrix (CTM) components.

This opaque protocol is not exportable across a SystemC interface.

2.7 Signaling protocols

Many components use the signaling protocols to indicate changes in state for signals such as interrupt and reset.

Each signaling protocol has two variants:

- One permits components to signal a state change to other components.
- One permits the other components to passively query the current state of the signal.

2.7.1 Signal protocol

The Signal protocol has the `setValue` behavior.

```
setValue(enum sg::Signal::State) : void  
    Indicates a state change.
```

Valid values for `sg::Signal::State` are:

- `sg::Signal::Set`.
- `sg::Signal::Clear`.

2.7.2 StateSignal protocol

The `StateSignal` protocol has the `setValue` and `getValue` behaviors.

```
setValue(enum sg::Signal::State): void  
    indicates a value change.
```

```
getValue() : sg::Signal::State  
    reads the current value.
```

2.7.3 Value protocol

The Value protocol has the setValue behavior.

setValue(uint32_t value)
indicates a state change.

2.7.4 Value_64 protocol

This protocol provides the rules to communicate with the TrustZone® Memory Adapter (TZMA) to signal the remapped range X.

setValue(uint64_t value)
Sets a 64-bit wide address to the slave port.

2.7.5 ValueState protocol

The ValueState protocol has the setValue and getValue behaviors.

setValue(uint32_t value): void
indicates a value change.

getValue() : uint32_t
reads the current value state.

3. Fast Models components

This chapter describes all model components in Fast Models, organized by component type.

For each component, the documentation includes notes about using the model, describes any deviations in the model from the Technical Reference Manual (TRM), and describes the ports and parameters.

3.1 Component differences

This topic lists the new and changed components in this release.

Differences between 11.25.15 and 11.26.8

Table 3-1: Components added

Component	Quality level
3.5.42 ARM CortexA520CT_CortexA725CT on page 1851	CortexA520 rOp1=rel, CortexA725 rOp0=pre
3.5.43 ARM CortexA520CT_CortexA725CT_CortexX925CT on page 1898	CortexA520 rOp1=rel, CortexA725 rOp0=pre, CortexX925 rOp0=pre
3.5.48 ARM CortexA725CT on page 2084	Preliminary support
3.5.49 ARM CortexA725CT_CortexX925CT on page 2113	CortexA725 rOp0=pre, CortexX925 rOp0=pre
3.5.74 ARM CortexX925CT on page 2580	Preliminary support
3.5.78 ARM NeoverseN3CT on page 2681	Preliminary support
3.4.2 ClockGate_cpp on page 163	N/A
3.4.3 ClockSelector_cpp on page 164	N/A
3.10.26 EthosU85 on page 3254	Preliminary support
3.9.2 FrequencyProbe_cpp on page 3031	N/A
3.6.22 Mali_G725 on page 2888	Preliminary support
3.6.18 Mali_G78AE on page 2877	Preliminary support
3.7.23 MemoryMappedGenericTimer_cpp on page 2955	N/A
3.7.25 MemoryMappedGenericWatchdog_cpp on page 2959	N/A
3.4.9 PLLControl_cpp on page 172	N/A
3.10.85 RSS_CPU_Private_Region on page 4044	Alpha support
3.4.10 ScalableClockControl_cpp on page 173	N/A
3.9.8 SignalInverter_cpp on page 3038	N/A
3.4.11 SwitchedClockControl_cpp on page 182	N/A
3.10.95 System_RAS_Agent on page 4115	Full support
3.7.59 Visualisation_sdl2_cpp on page 3017	N/A
3.9.12 WideAndGate_cpp on page 3041	N/A
3.9.13 WideOrGate_cpp on page 3041	N/A

Table 3-2: Components changed

Component	Has the IP revision changed?	Has the quality level changed?	Have ports been added or removed?	Have parameters been added or removed?
AEMv8RMPCT	No	No	No	Yes
AEMvACT	No	No	No	Yes
AMBAPV2PVBUS	No	No	No	Yes
AMBAPVACE2PVBUS	No	No	No	Yes
ARMCortexA510CT	No	Yes	No	Yes
ARMCortexA510CT_CortexA710CT	Yes	Yes	Yes	Yes
ARMCortexA510CT_CortexA710CT_CortexX2CT	Yes	Yes	Yes	Yes
ARMCortexA510CT_CortexA710CT_CortexX3CT	Yes	Yes	Yes	Yes
ARMCortexA510CT_CortexA715CT_CortexX3CT	Yes	Yes	Yes	Yes
ARMCortexA520AECT	No	No	Yes	Yes
ARMCortexA520CT	No	No	Yes	Yes
ARMCortexA520CT_CortexA720CT	Yes	Yes	Yes	Yes
ARMCortexA520CT_CortexA720CT_CortexX4CT	Yes	Yes	Yes	Yes
ARMCortexA55CT_CortexA75CT	No	No	No	Yes
ARMCortexA55CT_CortexA76CT	No	No	No	Yes
ARMCortexA55CT_CortexA78CT	No	No	No	Yes
ARMCortexA65AECT	No	No	No	Yes
ARMCortexA65AECT_CortexA76AECT	No	No	No	Yes
ARMCortexA65CT	No	No	No	Yes
ARMCortexA710CT	No	No	No	Yes
ARMCortexA715CT	No	No	No	Yes
ARMCortexA720AECT	No	No	Yes	Yes
ARMCortexA720CT	No	No	Yes	Yes
ARMCortexA76AECT	No	No	No	Yes
ARMCortexA76CT	No	No	No	Yes
ARMCortexA77CT	No	No	No	Yes
ARMCortexA78AECT	No	No	No	Yes
ARMCortexA78CCT	No	No	No	Yes
ARMCortexA78CT	No	No	No	Yes
ARMCortexM52CT	No	No	No	Yes
ARMCortexM55CT	No	No	No	Yes
ARMCortexM85CT	No	No	No	Yes
ARMCortexR82AECT	No	No	No	Yes
ARMCortexR82CT	Yes	Yes	No	Yes
ARMCortexX1CCT	No	No	No	Yes
ARMCortexX1CT	No	No	No	Yes
ARMCortexX2CT	No	No	No	Yes

Component	Has the IP revision changed?	Has the quality level changed?	Have ports been added or removed?	Have parameters been added or removed?
ARMCortexX3CT	No	No	No	Yes
ARMCortexX4CT	No	No	Yes	Yes
ARMNeoverseE1CT	No	No	No	Yes
ARMNeoverseN1CT	No	No	No	Yes
ARMNeoverseN2CT	No	No	No	Yes
ARMNeoverseV1CT	No	No	No	Yes
ARMNeoverseV2CT	No	No	No	Yes
ARMNeoverseV3AECT	No	No	Yes	Yes
ARMNeoverseV3CT	No	No	Yes	Yes
Base_PowerController	No	No	Yes	No
CMN700	No	No	No	Yes
GIC_IRI	No	No	No	Yes
GIC_IRI_Filter	No	No	No	Yes
IntegrityChecker	No	No	No	Yes
MMU_600	No	No	No	Yes
MMU_700	No	No	No	Yes
MMU_S3	Yes	Yes	Yes	Yes
Mali_C71	No	No	Yes	No
Mali_C78	Yes	No	Yes	No
MemoryMappedCounterModule	No	No	No	Yes
SecureCache	No	No	No	Yes
VirtioNetMMIO	No	No	No	Yes
VirtioRNG	No	No	No	Yes

3.2 Bridge components

This section describes the Bridge components.

These components allow conversion between the following protocols:

- PVBUS and AMBAPV.
- Signal and AMBAPVSignal.
- StateSignal and AMBAPVSignalState.
- Value(_64) and AMBAPVValue(64).
- ValueState(_64) and AMBAPVValueState(64).

LISA+ source for the bridge components is located in `$PVLIB_HOME/examples/SystemCEExport/Bridges/`.

The AMBAPV protocols and components are designed to interface with the AMBA® TLM PV library for ASI TLM 2.0. Fast Models provides this library as a standard way of mapping the AMBA protocol on top of ASI TLM 2.0.2 kit at PV level.

For more information about the AMBA TLM PV library for ASI TLM 2.0.2 kit, see the Fast Models documentation in `$MAXCORE_HOME/AMBA-PV/doc/`.

For more information about ASI TLM 2.0, see the Accellera documentation that is provided with the kit.

See also:

- [2.1 AMBA-PV protocols](#) on page 65
- [Accellera Systems Initiative](#)
- [AMBA-PV Extensions to TLM 2.0 Developer Guide](#)
- [Fast Models User Guide, SystemC Export with Multiple Instantiation](#)

3.2.1 AMBAPV2PVBUS

AMBA-PV to PVBUS protocol converter. This model is written in LISA+.

Changes in 11.26.8

Parameters added:

- `report_errors`

Iris and MTI instances for AMBAPV2PVBUS

This model has the following Iris instances:

Table 3-3: AMBAPV2PVBUS Iris instances

InstanceName	ComponentName
AMBAPV2PVBUS	AMBAPV2PVBUS
AMBAPV2PVBUS.bus_master	PVBUSMaster

This model has the following MTI trace components:

Table 3-4: AMBAPV2PVBUS MTI instances

InstanceName	ComponentName
AMBAPV2PVBUS	AMBAPV2PVBUS
AMBAPV2PVBUS.bus_master	PVBUSMaster

AMBAPV2PVBUS contains the following CADI targets:

- AMBAPV2PVBUS

About AMBAPV2PVBUS

- PVBUS does not support transactions with `byte_enable` set (strobing transactions, in AXI terms). This bridge component rejects them.
- Variants of this component also exist with multiple input and output ports.

Limitations

Fast Models bridges between PVBUS and AMBA-PV can transport Memory Tagging Extension (MTE) operations (tag stores, tag loads, and tag-checked loads and stores).

These operations are transported opaquely, so the endpoint must be using PVBUS. This means you cannot handle these operations in your own TLM components.

Ports for AMBAPV2PVBUS

Table 3-5: Ports

Name	Protocol	Type	Description
amba_pv_s	AMBAPV	Slave	-
pvbush_m	PVBUS	Master	-

Parameters for AMBAPV2PVBUS

base_addr

Base address

Type

int

Default value

0x0

report_errors

Report transactions which do not comply with PVBUS protocol requirements

Type

bool

Default value

0x0

shareable

Shareable default

Type

bool

Default value

0x1

3.2.2 AMBAPVACE2PVBUS

AMBA-PV ACE to PVBUS protocol converter. This model is written in LISA+.

Changes in 11.26.8

Parameters added:

- `report_errors`

Iris and MTI instances for AMBAPVACE2PVBUS

This model has the following Iris instances:

Table 3-6: AMBAPVACE2PVBUS Iris instances

InstanceName	ComponentName
AMBAPVACE2PVBUS	AMBAPVACE2PVBUS
AMBAPVACE2PVBUS.bus_master	PVBUSMaster

This model has the following MTI trace components:

Table 3-7: AMBAPVACE2PVBUS MTI instances

InstanceName	ComponentName
AMBAPVACE2PVBUS	AMBAPVACE2PVBUS
AMBAPVACE2PVBUS.bus_master	PVBUSMaster

AMBAPVACE2PVBUS contains the following CADI targets:

- AMBAPVACE2PVBUS

About AMBAPVACE2PVBUS

- AMBAPVACE2PVBUS depends on the AMBA-PV API, which must be at least version 1.4.
- The translation of bus transactions by the bridge has some impact on performance. Bus masters that cache memory transactions avoid much of this impact. The bridge does not support DMI.
- PVBUS does not support transactions with `byte_enable` set (strobing transactions, in AXI terms). This bridge component rejects them.

Ports for AMBAPVACE2PVBUS

Table 3-8: Ports

Name	Protocol	Type	Description
amba_pv_ace_s	AMBAPVACE	Slave	-
pvbust_m	PVBUS	Master	-

Parameters for AMBAPVACE2PVBUS

report_errors

Report transactions which do not comply with PVBUS protocol requirements

Type

bool

Default value

0x0

3.2.3 AMBAPVSignal2SGSignal

AMBA-PV Signal to SystemGenerator Signal protocol converter. This model is written in LISA+.

Iris and MTI instances for AMBAPVSignal2SGSignal

This model has the following Iris instances:

Table 3-9: AMBAPVSignal2SGSignal Iris instances

InstanceName	ComponentName
AMBAPVSignal2SGSignal	AMBAPVSignal2SGSignal



Variants of this component also exist with multiple input and output ports.

Ports for AMBAPVSignal2SGSignal

Table 3-10: Ports

Name	Protocol	Type	Description
amba_pv_signal_s	AMBAPVSignal	Slave	Input slave port for connection from top-level AMBAPVSignal slave port.
sg_signal_m	Signal	Master	Handles outgoing signal state changes. Converted signal state changes are sent out through this port.

3.2.4 AMBAPVSignalState2SGStateSignal

AMBA-PV SignalState to SystemGenerator StateSignal protocol converter. This model is written in LISA+.

Iris and MTI instances for AMBAPVSignalState2SGStateSignal

This model has the following Iris instances:

Table 3-11: AMBAPVSignalState2SGStateSignal Iris instances

InstanceName	ComponentName
AMBAPVSignalState2SGStateSignal	AMBAPVSignalState2SGStateSignal



Variants of this component also exist with multiple input and output ports.

Ports for AMBAPVSignalState2SGStateSignal

Table 3-12: Ports

Name	Protocol	Type	Description
amba_pv_signal_s	AMBAPVSignalState	Slave	-
sg_signal_m	StateSignal	Master	-

3.2.5 AMBAPVValue2SGValue

AMBA-PV Value to SystemGenerator Value protocol converter. This model is written in LISA+.

Iris and MTI instances for AMBAPVValue2SGValue

This model has the following Iris instances:

Table 3-13: AMBAPVValue2SGValue Iris instances

InstanceName	ComponentName
AMBAPVValue2SGValue	AMBAPVValue2SGValue



Variants of this component also exist with multiple input and output ports.

Ports for AMBAPVValue2SGValue

Table 3-14: Ports

Name	Protocol	Type	Description
amba_pv_value_s	AMBAPVValue	Slave	-
sg_value_m	Value	Master	-

3.2.6 AMBAPVValue2SGValue64

AMBA-PV Value64 to SystemGenerator Value_64 protocol converter. This model is written in LISA+.

Iris and MTI instances for AMBAPVValue2SGValue64

This model has the following Iris instances:

Table 3-15: AMBAPVValue2SGValue64 Iris instances

InstanceName	ComponentName
AMBAPVValue2SGValue64	AMBAPVValue2SGValue64



Variants of this component also exist with multiple input and output ports.

Ports for AMBAPVValue2SGValue64

Table 3-16: Ports

Name	Protocol	Type	Description
amba_pv_value_s	AMBAPVValue64	Slave	-
sg_value_m	Value_64	Master	-

3.2.7 AMBAPVValue642SMMUv3AEMIdentify

AMBA-PV Value64 to SMMUv3AEMIdentify protocol converter. This model is written in LISA+.

Iris and MTI instances for AMBAPVValue642SMMUv3AEMIdentify

This model has the following Iris instances:

Table 3-17: AMBAPVValue642SMMUv3AEMIdentify Iris instances

InstanceName	ComponentName
AMBAPVValue642SMMUv3AEMIdentify	AMBAPVValue642SMMUv3AEMIdentify

Ports for AMBAPVValue642SMMUv3AEMIdentify

Table 3-18: Ports

Name	Protocol	Type	Description
identify	SMMUv3AEMIdentifyProtocol	Master	-
identify_reply	AMBAPVValue64	Master	-
identify_request	AMBAPVValue64	Slave	-

3.2.8 AMBAPVValue642VECB

AMBA-PV to VECB protocol converter. This model is written in LISA+.

Iris and MTI instances for AMBAPVValue642VECB

This model has the following Iris instances:

Table 3-19: AMBAPVValue642VECB Iris instances

InstanceName	ComponentName
AMBAPVValue642VECB	AMBAPVValue642VECB

AMBAPVValue642VECB contains the following CADI targets:

- AMBAPVValue642VECB

Ports for AMBAPVValue642VECB

Table 3-20: Ports

Name	Protocol	Type	Description
amba_pv_ctrl_s	AMBAPVValue	Slave	-
amba_pv_data_s	AMBAPVValue64	Slave	-
vecb_m	VECBProtocol	Master	-

3.2.9 AMBAPVValueState2SGValueState

AMBA-PV ValueState to SystemGenerator ValueState protocol converter. This model is written in LISA+.

Iris and MTI instances for AMBAPVValueState2SGValueState

This model has the following Iris instances:

Table 3-21: AMBAPVValueState2SGValueState Iris instances

InstanceName	ComponentName
AMBAPVValueState2SGValueState	AMBAPVValueState2SGValueState



Note

Variants of this component also exist with multiple input and output ports.

Ports for AMBAPVValueState2SGValueState

Table 3-22: Ports

Name	Protocol	Type	Description
amba_pv_value_s	AMBAPVValueState	Slave	-
sg_value_m	ValueState	Master	-

3.2.10 AMBAPVValueState2SGValueState64

AMBA-PV ValueState64 to SystemGenerator ValueState_64 protocol converter. This model is written in LISA+.

Iris and MTI instances for AMBAPVValueState2SGValueState64

This model has the following Iris instances:

Table 3-23: AMBAPVValueState2SGValueState64 Iris instances

InstanceName	ComponentName
AMBAPVValueState2SGValueState64	AMBAPVValueState2SGValueState64



Variants of this component also exist with multiple input and output ports.

Ports for AMBAPVValueState2SGValueState64

Table 3-24: Ports

Name	Protocol	Type	Description
amba_pv_value_s	AMBAPVValueState64	Slave	-
sg_value_m	ValueState_64	Master	-

3.2.11 BroadcastSignal2AMBAPVSignal

Broadcast signal to AMBAPVSignal coverter. This model is written in LISA+.

Iris and MTI instances for BroadcastSignal2AMBAPVSignal

This model has the following Iris instances:

Table 3-25: BroadcastSignal2AMBAPVSignal Iris instances

InstanceName	ComponentName
BroadcastSignal2AMBAPVSignal	BroadcastSignal2AMBAPVSignal

BroadcastSignal2AMBAPVSignal contains the following CADI targets:

- BroadcastSignal2AMBAPVSignal

Ports for BroadcastSignal2AMBAPVSignal

Table 3-26: Ports

Name	Protocol	Type	Description
amba_pv_signal_m	AMBAPVSignal	Master	-
amba_pv_signal_s	AMBAPVSignal	Slave	-

Name	Protocol	Type	Description
b_signal	Signal	Broadcast	-

3.2.12 Clock2SystemC

Clock to SystemC Converter. This model is written in LISA+.

Iris and MTI instances for Clock2SystemC

This model has the following Iris instances:

Table 3-27: Clock2SystemC Iris instances

InstanceName	ComponentName
Clock2SystemC	Clock2SystemC

Clock2SystemC contains the following CADI targets:

- Clock2SystemC

Ports for Clock2SystemC

Table 3-28: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	-
current_ticks_s	AMBAPVValueState64	Slave	-
get_clock_s	AMBAPVValueState64	Slave	-
rate_in_Hz_s	AMBAPVValueState64	Slave	-
set_clock_m	AMBAPVValue64	Master	-

3.2.13 ClockRateConversion

ClockRateControl to rate in Hz (Value_64) Converter. This model is written in LISA+.

Iris and MTI instances for ClockRateConversion

This model has the following Iris instances:

Table 3-29: ClockRateConversion Iris instances

InstanceName	ComponentName
ClockRateConversion	ClockRateConversion
ClockRateConversion.clk_div0	ClockDivider
ClockRateConversion.clk_div1	ClockDivider
ClockRateConversion.clk_div2	ClockDivider
ClockRateConversion.clk_div3	ClockDivider

This model has the following MTI trace components:

Table 3-30: ClockRateConversion MTI instances

InstanceName	ComponentName
ClockRateConversion.clk_div0	ClockDivider
ClockRateConversion.clk_div1	ClockDivider
ClockRateConversion.clk_div2	ClockDivider
ClockRateConversion.clk_div3	ClockDivider

ClockRateConversion contains the following CADI targets:

- ClockDivider
- ClockRateConversion

Ports for ClockRateConversion

Table 3-31: Ports

Name	Protocol	Type	Description
clock	ClockSignal	Slave	-
rate_ctrl[4]	ClockRateControl	Slave	-
rate_hz[4]	Value_64	Master	-

Parameters for ClockRateConversion

clk_div0.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clk_div0.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clk_div1.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clk_div1.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clk_div2.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clk_div2.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clk_div3.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clk_div3.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

3.2.14 ClockSignal2SC_ClockSignal

ClockSignal to SystemC ClockSignal converter. This model is written in LISA+.

Iris and MTI instances for ClockSignal2SC_ClockSignal

This model has the following Iris instances:

Table 3-32: ClockSignal2SC_ClockSignal Iris instances

InstanceName	ComponentName
ClockSignal2SC_ClockSignal	ClockSignal2SC_ClockSignal

ClockSignal2SC_ClockSignal contains the following CADI targets:

- ClockSignal2SC_ClockSignal

Ports for ClockSignal2SC_ClockSignal

Table 3-33: Ports

Name	Protocol	Type	Description
clk_out	ClockSignal	Slave	-
sc_clk_out	SC_ClockSignal	Master	-

3.2.15 CoprocBus2SystemC

CoprocBusProtocol to SystemCCoprocBusProtocol converter. This model is written in LISA+.

Iris and MTI instances for CoprocBus2SystemC

This model has the following Iris instances:

Table 3-34: CoprocBus2SystemC Iris instances

InstanceName	ComponentName
CoprocBus2SystemC	CoprocBus2SystemC

CoprocBus2SystemC contains the following CADI targets:

- CoprocBus2SystemC

Ports for CoprocBus2SystemC

Table 3-35: Ports

Name	Protocol	Type	Description
coproc_bus_s	CoprocBusProtocol	Slave	-
sc_coproc_bus_m	SystemCCoprocBusProtocol	Master	-

3.2.16 CounterInterface2SystemC

CounterInterface to SystemC Converter. This model is written in LISA+.

Iris and MTI instances for CounterInterface2SystemC

This model has the following Iris instances:

Table 3-36: CounterInterface2SystemC Iris instances

InstanceName	ComponentName
CounterInterface2SystemC	CounterInterface2SystemC

CounterInterface2SystemC contains the following CADI targets:

- CounterInterface2SystemC

Ports for CounterInterface2SystemC

Table 3-37: Ports

Name	Protocol	Type	Description
amba_pv_eventUpdate_m	AMBAPVValue	Master	-
amba_pv_getCounterValue_s	AMBAPVValueState64	Slave	-
amba_pv_requestEventUpdate_s	AMBAPVValue64	Slave	-
amba_pv_requestSignalUpdate_s	AMBAPVValue64	Slave	-
amba_pv_setEnabled_m	AMBAPVValue	Master	-
amba_pv_signalUpdate_m	AMBAPVValue	Master	-
cntvalueb	CounterInterface	Slave	-

3.2.17 InstructionCount2SystemC

InstructionCount to SystemC Converter. This model is written in LISA+.

Iris and MTI instances for InstructionCount2SystemC

This model has the following Iris instances:

Table 3-38: InstructionCount2SystemC Iris instances

InstanceName	ComponentName
InstructionCount2SystemC	InstructionCount2SystemC

InstructionCount2SystemC contains the following CADI targets:

- InstructionCount2SystemC



Note

Variants of this component also exist with multiple input and output ports.

Ports for InstructionCount2SystemC

Table 3-39: Ports

Name	Protocol	Type	Description
inst_count	AMBAPVValueState64	Slave	-

Name	Protocol	Type	Description
run_state	AMBAPVValueState	Slave	-
ticks	InstructionCount	Slave	-

3.2.18 LCD2SystemC

Converts LCD protocol to SystemC. This model is written in LISA+.

Iris and MTI instances for LCD2SystemC

This model has the following Iris instances:

Table 3-40: LCD2SystemC Iris instances

InstanceName	ComponentName
LCD2SystemC	LCD2SystemC

LCD2SystemC contains the following CADI targets:

- LCD2SystemC

Ports for LCD2SystemC

Table 3-41: Ports

Name	Protocol	Type	Description
all_received_sPL	AMBAPVSignal	Master	-
all_received_u	AMBAPVSignal	Master	-
lcd_s	LCD	Slave	-
lock_m	AMBAPVValueState64	Master	-
setPreferredLayout_d	AMBAPVValue	Master	-
setPreferredLayout_h	AMBAPVValue	Master	-
setPreferredLayout_w	AMBAPVValue	Master	-
unlock_m	AMBAPVSignal	Master	-
update_h	AMBAPVValue	Master	-
update_w	AMBAPVValue	Master	-
update_x	AMBAPVValue	Master	-
update_y	AMBAPVValue	Master	-

3.2.19 PChannel2SystemC

PChannel to SystemC Converter. This model is written in LISA+.

Iris and MTI instances for PChannel2SystemC

This model has the following Iris instances:

Table 3-42: PChannel2SystemC Iris instances

InstanceName	ComponentName
PChannel2SystemC	PChannel2SystemC

PChannel2SystemC contains the following CADI targets:

- PChannel2SystemC

Ports for PChannel2SystemC

Table 3-43: Ports

Name	Protocol	Type	Description
pchannel	PChannel	Slave	-
sc_pchannel	SystemCPChannel	Master	-

3.2.20 PVBUS2AMBAPV

PVBus to AMBA-PV protocol converter. This model is written in LISA+.

Iris and MTI instances for PVBUS2AMBAPV

This model has the following Iris instances:

Table 3-44: PVBUS2AMBAPV Iris instances

InstanceName	ComponentName
PVBus2AMBAPV	PVBus2AMBAPV
PVBus2AMBAPV.bus_bridge	PVBusBridge

This model has the following MTI trace components:

Table 3-45: PVBUS2AMBAPV MTI instances

InstanceName	ComponentName
PVBus2AMBAPV	PVBus2AMBAPV
PVBus2AMBAPV.bus_bridge	PVBusBridge

PVBus2AMBAPV contains the following CADI targets:

- PVBus2AMBAPV

About PVBUS2AMBAPV



Note

Variants of PVBus2AMBAPV also exist with multiple input and output ports.

The AMBAPV protocol definition in LISA, `AMBAPVProtocol.lisa`, specifies a 64-bit bus width, so the PVBUS2AMBAPV bridge also handles a 64-bit bus width.

If you need to connect to a component that uses a bus interface with a smaller or larger bus width, the recommended method is to insert a downsizer or upsizer respectively.

Alternatively, you could define a new bus protocol with the required bit width, for example AMBAPV32, and update the corresponding bridges to use the new protocol on AMBA-PV ports:

```
master port<AMBAPV32> amba_pv_m
```

Limitations

Fast Models bridges between PVBUS and AMBA-PV can transport Memory Tagging Extension (MTE) operations (tag stores, tag loads, and tag-checked loads and stores).

These operations are transported opaquely, so the endpoint must be using PVBUS. This means you cannot handle these operations in your own TLM components.

Dumping the DMI cache

DMI viewer provides the debugging functionality of the PVBUS2AMBAPV bridge. When activated, it dumps the content of the DMI cache in the bridge in the following CSV format:

```
Range_start, Range_end_incl, Pointer, Latency, R/W, Attributes
```

To activate this functionality, a name for the counters output file must be set, using the `counters-file-name` parameter. If the counters file name is set, when `dump-dmi-cache` is set to 1 at runtime, the DMI cache of the bridge is dumped. The runtime parameter is always reset to 0 when the dump has completed.

Ports for PVBUS2AMBAPV

Table 3-46: Ports

Name	Protocol	Type	Description
<code>amba_pv_m</code>	AMBAPV	Master	-
<code>pvbuss_s</code>	PVBUS	Slave	-

Parameters for PVBUS2AMBAPV

counters-file-name

Prefix of the file name to store counters at the end of simulation

Type

string

Default value

dmi-container-type

Type of the DMI cache. Allowed values: TZAttr and FullAttr

Type

string

Default value

FullAttr

dump-dmi-cache

Dumps the content of the DMI cache into a file

Type

bool

Default value

0x0

dump-dmi-file-name

Prefix of the file name to dump the content of the DMI when requested

Type

string

Default value**force-dmi-size**

Force DMI start and end address to be 4kB-aligned

Type

bool

Default value

0x1

min-range-to-cache

Min DMI range size to cache in the bridge

Type

int

Default value

0x10000

size

Maximum size of memory region

Type

int

Default value

0x1000000000000

3.2.21 PVBUS2AMBAPVACE

PVBus to AMBA-PV ACE protocol converter. This model is written in LISA+.

Iris and MTI instances for PVBUS2AMBAPVACE

This model has the following Iris instances:

Table 3-47: PVBUS2AMBAPVACE Iris instances

InstanceName	ComponentName
PVBus2AMBAPVACE	PVBus2AMBAPVACE
PVBus2AMBAPVACE.bus_bridge	PVBusBridge
PVBus2AMBAPVACE.pvbus_tlm_switch	PVBusMapper

This model has the following MTI trace components:

Table 3-48: PVBUS2AMBAPVACE MTI instances

InstanceName	ComponentName
PVBus2AMBAPVACE	PVBus2AMBAPVACE
PVBus2AMBAPVACE.bus_bridge	PVBusBridge
PVBus2AMBAPVACE.pvbus_tlm_switch	PVBusMapper

PVBus2AMBAPVACE contains the following CADI targets:

- PVBus2AMBAPVACE

About PVBUS2AMBAPVACE

PVBus2AMBAPVACE depends on the AMBA-PV API, which must be at least version 1.4.

The translation of bus transactions by the bridge has some impact on performance. Bus masters that cache memory transactions avoid much of this impact.

DMI viewer provides the debugging functionality of the PVBus2AMBAPV bridge. When activated, it dumps the content of the DMI cache in the bridge in the following CSV format:

Range start, Range end incl, Pointer, Latency, R/W, Attributes

To activate this functionality, set a name for the counters output file using the `counters-file-name` parameter. If the counters file name is set, when `dump-dmi-cache` is set to 1 at runtime, the DMI cache of the bridge is dumped. The runtime parameter is always reset to 0 when the dump has completed.

Ports for PVBUS2AMBAPVACE

Table 3-49: Ports

Name	Protocol	Type	Description
amba_pv_ace_m	AMBAPVACE	Master	-
pvbus_over_tlm_control	PVBusOverTLMControl	Slave	-

Name	Protocol	Type	Description
pvbuss_s	PVBus	Slave	-

Parameters for PVBus2AMBAPVACE

counters-file-name

Prefix of the file name to store counters at the end of simulation

Type

string

Default value

dmi-cache-name

DEPRECATED: This parameter will be ignored. Name of the DMI cache. Useful for multiple bridges to share the same cache

Type

string

Default value

dmi-container-type

Type of the DMI cache. Allowed values: TZAttr and FullAttr. When there are caches downstream of this bridge and cache-state-modelled=1 and route-tlm=1 use only FullAttr.

Type

string

Default value

FullAttr

dump-dmi-cache

Dumps the content of the DMI cache into a file

Type

bool

Default value

0x0

dump-dmi-file-name

Prefix of the file name to dump the content of the DMI when requested

Type

string

Default value

force-dmi-size

Force DMI start and end address to be 4kB-aligned

Type

bool

Default value

0x1

min-range-to-cache

Min DMI range size to cache in the bridge

Type

int

Default value

0x10000

route-tlm

Route all the PVBUS traffic explicitly to the TLM bus. Allows to monitor transactions on the TLM bus but slows down the emulation. The routing must always be to TLM if there is not a corresponding AMBAPVACE2PVBUS bridge downstream.

Type

bool

Default value

0x1

route-tlm-filter

Route TLM filter set a range (or multiple ranges) of addresses that will use PVBUS even if route-tlm is set to true. The route-tlm-filter is specified in JSON format. Example, [{"begin":0x2f000000, "size":0x1000}, {"begin":0x4f000000, "size":0x2000}].

Type

string

Default value**set-ace-lite**

Set bridge mode when connecting to ace-lite ports. If true, the bridge will not deal with SNOOPS.

Type

bool

Default value

0x0

size

Maximum size of memory region, i.e. the first unsupported address.

Type

int

Default value

0x10000000000000

3.2.22 PVBUSBridge

A PVBUSBridge bridges incoming transactions to a PVDevice port. This model is written in C++.

Iris and MTI instances for PVBUSBridge

This model has the following Iris instances:

Table 3-50: PVBUSBridge Iris instances

InstanceName	ComponentName
PVBUSBridge	PVBUSBridge

This model has the following MTI trace components:

Table 3-51: PVBUSBridge MTI instances

InstanceName	ComponentName
PVBUSBridge	PVBUSBridge

Ports for PVBUSBridge**Table 3-52: Ports**

Name	Protocol	Type	Description
control	PVBUSBridgeControl	Slave	Control signal.
device	PVDevice	Master	Optimised connection out to devices.
dump_dmi	Signal	Slave	On the assert of this signal the bridge will dump dmi cache content into a csv file
pvbuss	PVBUS	Slave	Connection in from bus master.
reset	Signal	Slave	On the assert of this signal, a reset of the bus slave will be latched this is used by the bus deadlock detection logic.

3.2.23 SC_ClockSignal2ClockSignal

SystemC ClockSignal to ClockSignal converter. This model is written in LISA+.

Iris and MTI instances for SC_ClockSignal2ClockSignal

This model has the following Iris instances:

Table 3-53: SC_ClockSignal2ClockSignal Iris instances

InstanceName	ComponentName
SC_ClockSignal2ClockSignal	SC_ClockSignal2ClockSignal

SC_ClockSignal2ClockSignal contains the following CADI targets:

- SC_ClockSignal2ClockSignal

Ports for SC_ClockSignal2ClockSignal

Table 3-54: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Master	-
sc_clk_in	SC_ClockSignal	Slave	-

3.2.24 SGSignal2AMBAPVSignal

SystemGenerator Signal to AMBA-PV Signal protocol converter. This model is written in LISA+.

Iris and MTI instances for SGSignal2AMBAPVSignal

This model has the following Iris instances:

Table 3-55: SGSignal2AMBAPVSignal Iris instances

InstanceName	ComponentName
SGSignal2AMBAPVSignal	SGSignal2AMBAPVSignal



Variants of this component also exist with multiple input and output ports.

Ports for SGSignal2AMBAPVSignal

Table 3-56: Ports

Name	Protocol	Type	Description
amba_pv_signal_m	AMBAPVSignal	Master	Output master port for connection to top-level AMBAPVSignal master port. Converted signal state changes are sent out through this port.
sg_signal_s	Signal	Slave	Handles incoming signal state changes.

3.2.25 SGStateSignal2AMBAPVSignalState

SystemGenerator StateSignal to AMBA-PV SignalState protocol converter. This model is written in LISA+.

Iris and MTI instances for SGStateSignal2AMBAPVSignalState

This model has the following Iris instances:

Table 3-57: SGStateSignal2AMBAPVSignalState Iris instances

InstanceName	ComponentName
SGStateSignal2AMBAPVSignalState	SGStateSignal2AMBAPVSignalState



Variants of this component also exist with multiple input and output ports.

Ports for SGStateSignal2AMBAPVSignalState

Table 3-58: Ports

Name	Protocol	Type	Description
amba_pv_signal_m	AMBAPVSignalState	Master	Output master port for connection to top-level AMBAPVValue64 master port. Converted value changes are sent out through this port.
sg_signal_s	StateSignal	Slave	Handles incoming value changes.

3.2.26 SGValue2AMBAPVValue

SystemGenerator Value to AMBA-PV Value protocol converter. This model is written in LISA+.

Iris and MTI instances for SGValue2AMBAPVValue

This model has the following Iris instances:

Table 3-59: SGValue2AMBAPVValue Iris instances

InstanceName	ComponentName
SGValue2AMBAPVValue	SGValue2AMBAPVValue



Variants of this component also exist with multiple input and output ports.

Ports for SGValue2AMBAPVValue

Table 3-60: Ports

Name	Protocol	Type	Description
amba_pv_value_m	AMBAPVValue	Master	Output master port for connection to top-level AMBAPVValue master port. Converted value changes are sent out through this port.
sg_value_s	Value	Slave	Handles incoming value changes.

3.2.27 SGValue2AMBAPVValue64

SystemGenerator Value_64 to AMBA-PV Value64 protocol converter. This model is written in LISA+.

Iris and MTI instances for SGValue2AMBAPVValue64

This model has the following Iris instances:

Table 3-61: SGValue2AMBAPVValue64 Iris instances

InstanceName	ComponentName
SGValue2AMBAPVValue64	SGValue2AMBAPVValue64



Variants of this component also exist with multiple input and output ports.

Ports for SGValue2AMBAPVValue64

Table 3-62: Ports

Name	Protocol	Type	Description
amba_pv_value_m	AMBAPVValue64	Master	Output master port for connection to top-level AMBAPVValue master port. Converted value changes are sent out through this port.
sg_value_s	Value_64	Slave	Handles incoming value changes.

3.2.28 SGValueState2AMBAPVValueState

SystemGenerator ValueState to AMBA-PV ValueState protocol converter. This model is written in LISA+.

Iris and MTI instances for SGValueState2AMBAPVValueState

This model has the following Iris instances:

Table 3-63: SGValueState2AMBAPVValueState Iris instances

InstanceName	ComponentName
SGValueState2AMBAPVValueState	SGValueState2AMBAPVValueState



Variants of this component also exist with multiple input and output ports.

Ports for SGValueState2AMBAPVValueState

Table 3-64: Ports

Name	Protocol	Type	Description
amba_pv_value_m	AMBAPVValueState	Master	Output master port for connection to top-level AMBAPVValue master port. Converted value changes are sent out through this port.
sg_value_s	ValueState	Slave	Handles incoming value changes.

3.2.29 SGValueState2AMBAPVValueState64

SystemGenerator ValueState_64 to AMBA-PV ValueState64 protocol converter. This model is written in LISA+.

Iris and MTI instances for SGValueState2AMBAPVValueState64

This model has the following Iris instances:

Table 3-65: SGValueState2AMBAPVValueState64 Iris instances

InstanceName	ComponentName
SGValueState2AMBAPVValueState64	SGValueState2AMBAPVValueState64



Variants of this component also exist with multiple input and output ports.

Ports for SGValueState2AMBAPVValueState64

Table 3-66: Ports

Name	Protocol	Type	Description
amba_pv_value_m	AMBAPVValueState64	Master	Output master port for connection to top-level AMBAPVValue64 master port. Converted value changes are sent out through this port.
sg_value_s	ValueState_64	Slave	Handles incoming value changes.

3.2.30 SMMUv3AEMIdentify2AMBAPVValue64

SMMUv3AEMIdentify to AMBA-PV Value64 protocol converter. This model is written in LISA+.

Iris and MTI instances for SMMUv3AEMIdentify2AMBAPVValue64

This model has the following Iris instances:

Table 3-67: SMMUv3AEMIdentify2AMBAPVValue64 Iris instances

InstanceName	ComponentName
SMMUv3AEMIdentify2AMBAPVValue64	SMMUv3AEMIdentify2AMBAPVValue64

Ports for SMMUv3AEMIdentify2AMBAPVValue64

Table 3-68: Ports

Name	Protocol	Type	Description
identify	SMMUv3AEMIdentifyProtocol	Slave	SMMUv3AEMIdentifyProtocol input.
identify_reply	AMBAPVValue64	Slave	From SystemC.
identify_request	AMBAPVValue64	Master	To SystemC.

3.2.31 SystemC2Clock

Clock to SystemC Converter. This model is written in LISA+.

Iris and MTI instances for SystemC2Clock

This model has the following Iris instances:

Table 3-69: SystemC2Clock Iris instances

InstanceName	ComponentName
SystemC2Clock	SystemC2Clock

SystemC2Clock contains the following CADI targets:

- SystemC2Clock

Ports for SystemC2Clock

Table 3-70: Ports

Name	Protocol	Type	Description
clk_out	ClockSignal	Master	ClockSignal output
current_ticks_m	AMBAPVValueState64	Master	To SystemC.
get_clock_m	AMBAPVValueState64	Master	To SystemC.
rate_in_Hz_m	AMBAPVValueState64	Master	To SystemC.
set_clock_s	AMBAPVValue64	Slave	From SystemC.

3.2.32 SystemC2CprocBus

SystemCCoprocBusProtocol to CoprocBusProtocol converter. This model is written in LISA+.

Iris and MTI instances for SystemC2CprocBus

This model has the following Iris instances:

Table 3-71: SystemC2CprocBus Iris instances

InstanceName	ComponentName
SystemC2CprocBus	SystemC2CprocBus

SystemC2CprocBus contains the following CADI targets:

- SystemC2CprocBus

Ports for SystemC2CprocBus

Table 3-72: Ports

Name	Protocol	Type	Description
coproc_bus_m	CoprocBusProtocol	Master	-
sc_coproc_bus_s	SystemCCoprocBusProtocol	Slave	-

3.2.33 SystemC2CounterInterface

SystemC to CounterInterface Converter. This model is written in LISA+.

Iris and MTI instances for SystemC2CounterInterface

This model has the following Iris instances:

Table 3-73: SystemC2CounterInterface Iris instances

InstanceName	ComponentName
SystemC2CounterInterface	SystemC2CounterInterface

SystemC2CounterInterface contains the following CADI targets:

- SystemC2CounterInterface

Ports for SystemC2CounterInterface

Table 3-74: Ports

Name	Protocol	Type	Description
amba_pv_eventUpdate_s	AMBAPVValue	Slave	From SystemC.
amba_pv_getCounterValue_m	AMBAPVValueState64	Master	To SystemC.
amba_pv_requestEventUpdate_m	AMBAPVValue64	Master	To SystemC.
amba_pv_requestSignalUpdate_m	AMBAPVValue64	Master	To SystemC.
amba_pv_setEnabled_s	AMBAPVValue	Slave	From SystemC.
amba_pv_signalUpdate_s	AMBAPVValue	Slave	From SystemC.
cntvalueb	CounterInterface	Master	-

3.2.34 SystemC2InstructionCount

SystemC to InstructionCount Converter. This model is written in LISA+.

Iris and MTI instances for SystemC2InstructionCount

This model has the following Iris instances:

Table 3-75: SystemC2InstructionCount Iris instances

InstanceName	ComponentName
SystemC2InstructionCount	SystemC2InstructionCount

SystemC2InstructionCount contains the following CADI targets:

- SystemC2InstructionCount



Variants of this component also exist with multiple input and output ports.

Ports for SystemC2InstructionCount

Table 3-76: Ports

Name	Protocol	Type	Description
inst_count	AMBAPVValueState64	Master	To SystemC to request instruction count.
run_state	AMBAPVValueState	Master	To SystemC to request run state.
ticks	InstructionCount	Master	InstructionCount input.

3.2.35 SystemC2LCD

Converts SystemC to LCD protocol. This model is written in LISA+.

Iris and MTI instances for SystemC2LCD

This model has the following Iris instances:

Table 3-77: SystemC2LCD Iris instances

InstanceName	ComponentName
SystemC2LCD	SystemC2LCD

SystemC2LCD contains the following CADI targets:

- SystemC2LCD

Ports for SystemC2LCD

Table 3-78: Ports

Name	Protocol	Type	Description
all_received_sPL	AMBAPVSignal	Slave	From SystemC.
all_received_u	AMBAPVSignal	Slave	From SystemC.
lcd_m	LCD	Master	LCD output.
lock_s	AMBAPVValueState64	Slave	From SystemC.
setPreferredLayout_d	AMBAPVValue	Slave	From SystemC.
setPreferredLayout_h	AMBAPVValue	Slave	From SystemC.
setPreferredLayout_w	AMBAPVValue	Slave	From SystemC.
unlock_s	AMBAPVSignal	Slave	From SystemC.
update_h	AMBAPVValue	Slave	From SystemC.
update_w	AMBAPVValue	Slave	From SystemC.
update_x	AMBAPVValue	Slave	From SystemC.
update_y	AMBAPVValue	Slave	From SystemC.

3.2.36 SystemC2PChannel

SystemC to PChannel Converter. This model is written in LISA+.

Iris and MTI instances for SystemC2PChannel

This model has the following Iris instances:

Table 3-79: SystemC2PChannel Iris instances

InstanceName	ComponentName
SystemC2PChannel	SystemC2PChannel

SystemC2PChannel contains the following CADI targets:

- SystemC2PChannel

Ports for SystemC2PChannel

Table 3-80: Ports

Name	Protocol	Type	Description
pchannel	PChannel	Master	-
sc_pchannel	SystemCPChannel	Slave	-

3.2.37 SystemC2VirtualEthernet

SystemC to VirtualEthernet Converter. This model is written in LISA+.

Iris and MTI instances for SystemC2VirtualEthernet

This model has the following Iris instances:

Table 3-81: SystemC2VirtualEthernet Iris instances

InstanceName	ComponentName
SystemC2VirtualEthernet	SystemC2VirtualEthernet

SystemC2VirtualEthernet contains the following CADI targets:

- SystemC2VirtualEthernet

Ports for SystemC2VirtualEthernet

Table 3-82: Ports

Name	Protocol	Type	Description
virtualethernet_m	VirtualEthernet	Master	-
virtualethernet_s	SC_VirtualEthernet	Slave	-

3.2.38 SystemC2v7VGICConfig

Converts SystemC to v7_vgic_configuration_protocol. This model is written in LISA+.

Iris and MTI instances for SystemC2v7VGICConfig

This model has the following Iris instances:

Table 3-83: SystemC2v7VGICConfig Iris instances

InstanceName	ComponentName
SystemC2v7VGICConfig	SystemC2v7VGICConfig

SystemC2v7VGICConfig contains the following CADI targets:

- SystemC2v7VGICConfig

Ports for SystemC2v7VGICConfig

Table 3-84: Ports

Name	Protocol	Type	Description
all_received_s	AMBAPVSignal	Slave	From SystemC.
cpu_interface_number_s	AMBAPVValue64	Slave	From SystemC.
inout_cluster_number_s	AMBAPVValue64	Slave	From SystemC.
inout_cpu_number_in_cluster_s	AMBAPVValue64	Slave	From SystemC.
master_id_mask_s	AMBAPVValue	Slave	From SystemC.
master_id_s	AMBAPVValue	Slave	From SystemC.
number_of_cores_s	AMBAPVValueState	Slave	From SystemC.
response_m	AMBAPVSignal	Master	To SystemC.
v7_vgic_config_m	v7_VGIC_Configuration_Protocol	Master	v7_VGIC_Configuration_Protocol output.

3.2.39 VECB2AMBAPVValue64

VECB protocol to AMBA-PV protocol converter. This model is written in LISA+.

Iris and MTI instances for VECB2AMBAPVValue64

This model has the following Iris instances:

Table 3-85: VECB2AMBAPVValue64 Iris instances

InstanceName	ComponentName
VECB2AMBAPVValue64	VECB2AMBAPVValue64

VECB2AMBAPVValue64 contains the following CADI targets:

- VECB2AMBAPVValue64

Ports for VECB2AMBAPVValue64

Table 3-86: Ports

Name	Protocol	Type	Description
amba_pv_ctrl_m	AMBAPVValue	Master	AMBAPV portout.
amba_pv_data_m	AMBAPVValue64	Master	AMBAPV portout.
vecb_s	VECBProtocol	Slave	VECB port in.

3.2.40 VirtualEthernet2SystemC

VirtualEthernet to SystemC Converter. This model is written in LISA+.

Iris and MTI instances for VirtualEthernet2SystemC

This model has the following Iris instances:

Table 3-87: VirtualEthernet2SystemC Iris instances

InstanceName	ComponentName
VirtualEthernet2SystemC	VirtualEthernet2SystemC

VirtualEthernet2SystemC contains the following CADI targets:

- VirtualEthernet2SystemC

Ports for VirtualEthernet2SystemC

Table 3-88: Ports

Name	Protocol	Type	Description
virtualethernet_m	SC_VirtualEthernet	Master	-
virtualethernet_s	VirtualEthernet	Slave	-

3.2.41 v7VGICConfig2SystemC

Converts v7_vgic_configuration_protocol to SystemC. This model is written in LISA+.

Iris and MTI instances for v7VGICConfig2SystemC

This model has the following Iris instances:

Table 3-89: v7VGICConfig2SystemC Iris instances

InstanceName	ComponentName
v7VGICConfig2SystemC	v7VGICConfig2SystemC

v7VGICConfig2SystemC contains the following CADI targets:

- v7VGICConfig2SystemC

Ports for v7VGICConfig2SystemC

Table 3-90: Ports

Name	Protocol	Type	Description
all_received	AMBAPVSignal	Master	Called when all other values have been set in opposite bridge.
cpu_interface_number_m	AMBAPVValue64	Master	To SystemC.
inout_cluster_number_m	AMBAPVValue64	Master	To SystemC.
inout_cpu_number_in_cluster_m	AMBAPVValue64	Master	To SystemC.
master_id_m	AMBAPVValue	Master	To SystemC.
master_id_mask_m	AMBAPVValue	Master	To SystemC.
number_of_cores_m	AMBAPVValueState	Master	To SystemC.
response_s	AMBAPVSignal	Slave	From SystemC.
v7_vgic_config_s	v7_VGIC_Configuration_Protocol	Slave	v7_VGIC_Configuration_Protocol input.

3.3 Bus components

This section describes the Bus components.

PVBus is the bus protocol that is used to model all memory-like buses in Fast Models. PVBus is an internal protocol. The PVBus components, which abstract the internal details, are the interface to the PVBus API.

PVBus components provide functionally accurate communication between bus masters and slaves. They are not software implementations of specific hardware, but instead are abstract components that are required by the software model environment.

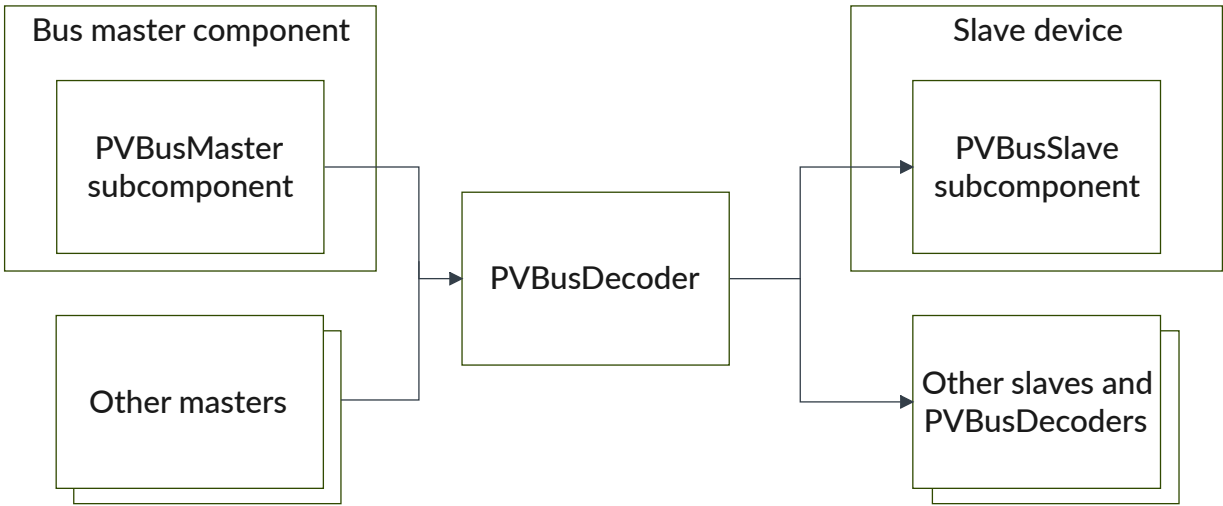
There is no modeling of handshaking or cycle counts. By removing this level of detail, and by using efficient internal communication mechanisms, PVBus components can provide very fast modeling of bus accesses. You must use these components correctly to achieve high simulation speeds.

Each bus master must contain a `PVBusMaster` subcomponent, and each bus slave must contain a `PVBusSlave` subcomponent. These subcomponents provide `PVBus` master and slave ports. Each `PVBus` master port can only connect to one slave, but any number of other masters can connect to the same slave. `PVBusDecoder`, `PVBusMaster` and `PVBusSlave` components communicate using the `PVBus` protocol.

`PVBusDecoder` components can be added to the bus system. Each of these permits its masters to connect to multiple slaves, each associated with a different bus address range.

`PVBusSlave` subcomponents provide built-in support for declaring memory-like, device-like, abort or ignore address ranges. `PVBus` has support for dealing efficiently with memory-like devices such as RAM, ROM, and Flash.

Figure 3-1: Sample bus topology



All communication over the `PVBUS` is performed using transactions that are generated by `PVBUSMaster` subcomponents and fulfilled by `PVBUSSlave` components. Transactions have a 32-bit Master ID, which is the ID of the bus master. Transactions can be routed to the slave device through its `PVBUSSlave` subcomponent. When configured, the `PVBUSSlave` can handle memory-like transactions efficiently without having to route these transactions to the slave device. Transactions are atomic unless slave devices block transactions, for example an SMMU with stall mode enabled. A slave device that can block transactions and all its upstream bus components must be re-entrant safe for bus transactions.

Fast Models provides some example `PVBUS` systems:

- `$PVLIB_HOME/examples/LISA/common/LISA/RemapDecoder.lisa`. This example dynamically modifies routing of requests based on a remap signal, using the `tzswitch` component.
- The directory `$PVLIB_HOME/examples/LISA/BusComponents/` contains a set of example components that show various ways of using the `PVBUS` interface.

3.3.1 Labeller

This model is written in LISA+.

Iris and MTI instances for Labeller

This model has the following Iris instances:

Table 3-91: Labeller Iris instances

InstanceName	ComponentName
Labeller	Labeller
Labeller.pvbusmodifier	PVBUSMapper

This model has the following MTI trace components:

Table 3-92: Labeller MTI instances

InstanceName	ComponentName
Labeller.pvbusmodifier	PVBusMapper

Labeller contains the following CADI targets:

- Labeller

About Labeller

Labeller and LabellerForDMA330 are utility components that allow the system designer to embed values into the Label field for transactions generated by a Bus Master. They are located between PVBus Master and Slave ports.

The following example creates a labeller to add an ID for an HDLCD controller that is upstream of a TZC_400. The system designer specifies a unique set of IDs for use as *Non-Secure Access IDs* (NSAIDs) in the TZC_400. The labeller can insert these IDs directly into the transaction.

```
p1370_hdlcd : PL370 HDLCD();
hdlcd_labeller : Labeller( "label" = 2 );
p1370_hdlcd.pvbus_m => hdlcd_labeller.pvbus_s;
hdlcd_labeller.pvbus_m => output_bus.pvbus_s;
```

Ports for Labeller

Table 3-93: Ports

Name	Protocol	Type	Description
pvbus_m	PVBus	Master	Output with modified MasterID.
pvbus_s	PVBus	Slave	Unmodified input.

Parameters for Labeller

label

The label to apply to all transactions flowing through the labeller

Type

int

Default value

0x0

3.3.2 LabellerForDMA330

This model is written in LISA+.

Iris and MTI instances for LabellerForDMA330

This model has the following Iris instances:

Table 3-94: LabellerForDMA330 Iris instances

InstanceName	ComponentName
LabellerForDMA330	LabellerForDMA330
LabellerForDMA330.pvbusmapper	PVBusMapper

This model has the following MTI trace components:

Table 3-95: LabellerForDMA330 MTI instances

InstanceName	ComponentName
LabellerForDMA330.pvbusmapper	PVBusMapper

LabellerForDMA330 contains the following CADI targets:

- LabellerForDMA330

About LabellerForDMA330

Labeller and LabellerForDMA330 are utility components that allow the system designer to embed values into the Label field for transactions generated by a Bus Master. They are located between PVBus Master and Slave ports.

Ports for LabellerForDMA330

Table 3-96: Ports

Name	Protocol	Type	Description
pvbus_m	PVBus	Master	Output with modified MasterID.
pvbus_s	PVBus	Slave	Unmodified input.

Parameters for LabellerForDMA330

dma330_data_label

The label to apply to all _data_ transactions flowing through the labeller. Used as a base value in conjunction with the channel ID if data-channel discrimination is enabled.

Type

int

Default value

0x0

dma330_discriminate_data_channels

Discriminate between DMA-330 data channels. Channel ID is added to the data label.

Type

bool

Default value

0x0

dma330_ns_instruction_label
The label to apply to all non-secure `_instructions_` transactions flowing through the labeller

Type
int

Default value
0x0

dma330_s_instruction_label
The label to apply to all secure `_instructions_` transactions flowing through the labeller

Type
int

Default value
0x0

3.3.3 LabellerForGPUProtMode

This model is written in LISA+.

Iris and MTI instances for LabellerForGPUProtMode

This model has the following Iris instances:

Table 3-97: LabellerForGPUProtMode Iris instances

InstanceName	ComponentName
LabellerForGPUProtMode	LabellerForGPUProtMode
LabellerForGPUProtMode.pvbusmodifier	PVBusMapper

This model has the following MTI trace components:

Table 3-98: LabellerForGPUProtMode MTI instances

InstanceName	ComponentName
LabellerForGPUProtMode.pvbusmodifier	PVBusMapper

LabellerForGPUProtMode contains the following CADI targets:

- LabellerForGPUProtMode

About LabellerForGPUProtMode

This component adds Non-Secure Access IDs (NSAIDs) to the transactions generated by the GPU. The NSAID is a four-bit number. It allows other components, such as a TrustZone® Controller (TZC) or a Dynamic Memory Controller (DMC) to filter transactions and control access to memory regions that are designated as protected.

Ports for LabellerForGPUProtMode

Table 3-99: Ports

Name	Protocol	Type	Description
prot_mode	Signal	Slave	Input to determine whether output is supposed to be protected or not
pvbus_m	PVBus	Master	Output with modified MasterID.
pvbus_s	PVBus	Slave	Unmodified input.

Parameters for LabellerForGPUProtMode

gpu_id_normal

NSAID to apply to all transactions flowing through the labeller when prot_mode is low.

Type

int

Default value

0x0

gpu_id_protected

NSAID to apply to all transactions flowing through the labeller when prot_mode is high.

Type

int

Default value

0x0

3.3.4 MSIRewriter

Recognise writes to the GITS_TRANSLATER register and rewrite them to go to the GITS_TRANSLATE64R register. The DeviceID is expected to be in the bottom 32 bits of ExtendedID of the transaction. Debug transactions and reads are not rewritten. This component can also, optionally, apply a 16 bit 'label' to the top 16 bits of the MasterID in the same way that the Labeller component does. This model is written in LISA+.

Iris and MTI instances for MSIRewriter

This model has the following Iris instances:

Table 3-100: MSIRewriter Iris instances

InstanceName	ComponentName
MSIRewriter	MSIRewriter
MSIRewriter.mapper	PVBusMapper
MSIRewriter.pvbustmaster	PVBusMaster
MSIRewriter.pvbusslave	PVBusSlave

This model has the following MTI trace components:

Table 3-101: MSIRewriter MTI instances

InstanceName	ComponentName
MSIRewriter	MSIRewriter
MSIRewriter.mapper	PVBusMapper
MSIRewriter.pvbusmaster	PVBusMaster
MSIRewriter.pvbuslave	PVBusSlave

MSIRewriter contains the following CADI targets:

- MSIRewriter

About MSIRewriter

MSIRewriter is a component that implements the functionality of the MSI-64 Encapsulator in GIC IP, for example GIC-700. For more information about GIC-700, see [GIC-700 Technical Reference Manual](#). For the GIC architecture specification version 3 and version 4, see [GIC Architecture Specification](#).

If an MSIRewriter component is used, it converts writes to the GITS_TRANSLATER register to writes to a model-only register called GITS_TRANSLATE64R. GITS_TRANSLATE64R holds the DeviceID in the upper 32 bits and the EventID in the lower 32 bits. The lower 32 bits of GITS_TRANSLATE64R correspond to the GITS_TRANSLATER register.

MasterID, ExtendedID, and UserFlags

These tables show how this model encodes information in the `MasterID`, `ExtendedID`, and `UserFlags` bus attributes:

Table 3-102: pvbus_s port

PVBus attribute	Bits used	Property encoded	Notes
MasterID	-	-	Not used.
ExtendedID	Bits[63:32]	Stream ID	-
	Bits[31:0]	Device ID	-
UserFlags	-	-	Not used.

Table 3-103: pvbus_m port

PVBus attribute	Bits used	Property encoded	Notes
MasterID	Bits[31:16]	Optional 16-bit label.	For usage, see the <code>label</code> parameter.
ExtendedID	-	-	Not used.
UserFlags	-	-	Not used.

Ports for MSIRewriter

Table 3-104: Ports

Name	Protocol	Type	Description
pvbus_m	PVBus	Master	-
pvbus_s	PVBus	Slave	-

Parameters for MSIRewriter

GITS_TRANSLATE64R_OFFSET

It is an offset from ITS0-Base.

Type

int

Default value

0x10048

ITS0-base

Register base address for ITS0. This base address is used to recognise writes to the GITS_TRANSLATER register within the ITS0's register frame.

Type

int

Default value

0x0

enable_rewriting

Enable rewriting.

Type

bool

Default value

0x1

label

If $< 2^{*16}$ then this is a label that is put in the top 16 bits of MasterID in the same way that the component Labeller does. This labelling is not controlled by enable_rewriting and is performed on all transactions (even rewritten ones)

Type

int

Default value

0xffffffff

log

Log level, 0 is off.

Type

int

Default value

0x0

3.3.5 PASSwitch

Allow transactions from Realm Management Extension(RME) worlds (realm/root/secure/non_secure) to be routed separately. This model is written in C++.

Ports for PASSwitch

Table 3-105: Ports

Name	Protocol	Type	Description
control	PASSwitchControl	Slave	Controls routing of transactions
pvbus_m[4]	PVBus	Master	Manager ports of PASSwitch
pvbus_s	PVBus	Slave	Subordinate port of PASSwitch

Parameters for PASSwitch

non_secure_port_index

Port Index for Secure world transactions to exit or -2 for IGNORE or -1 for ABORT [Default Index of 0]

Type

int

Default value

0x1

port_map_json

A JSON value describing ports for different address regions. The begin address and size values should be aligned to 4KiB. The format is as follows:{{ "begin": 0x0, "size": 0x1000, "port": 0 },{ "begin": 0x20000, "size": 0x50000, "port": 2 }}

Type

string

Default value

realm_port_index

Port Index for Realm world transactions to exit or -2 for IGNORE or -1 for ABORT [Default Index of 0]

Type

int

Default value

0x3

root_port_index

Port Index for Root world transactions to exit or -2 for IGNORE or -1 for ABORT [Default Index of 0]

Type

int

Default value

0x2

secure_port_index

Port Index for Secure world transactions to exit or -2 for IGNORE or -1 for ABORT [Default Index of 0]

Type

int

Default value

0x0

3.3.6 PVBUS4KBTo1KBSplitter

Takes 4KB of address range input on slave port and routes each 1KB to four different master ports. This model is written in LISA+.

Iris and MTI instances for PVBUS4KBTo1KBSplitter

This model has the following Iris instances:

Table 3-106: PVBUS4KBTo1KBSplitter Iris instances

InstanceName	ComponentName
PVBUS4KBTo1KBSplitter	PVBUS4KBTo1KBSplitter
PVBUS4KBTo1KBSplitter.input_slave	PVBUSSlave
PVBUS4KBTo1KBSplitter.output_master0	PVBUSMaster
PVBUS4KBTo1KBSplitter.output_master1	PVBUSMaster
PVBUS4KBTo1KBSplitter.output_master2	PVBUSMaster
PVBUS4KBTo1KBSplitter.output_master3	PVBUSMaster

This model has the following MTI trace components:

Table 3-107: PVBUS4KBTo1KBSplitter MTI instances

InstanceName	ComponentName
PVBUS4KBTo1KBSplitter.input_slave	PVBUSSlave
PVBUS4KBTo1KBSplitter.output_master0	PVBUSMaster
PVBUS4KBTo1KBSplitter.output_master1	PVBUSMaster
PVBUS4KBTo1KBSplitter.output_master2	PVBUSMaster
PVBUS4KBTo1KBSplitter.output_master3	PVBUSMaster

PVBUS4KBTo1KBSplitter contains the following CADI targets:

- PVBUS4KBTo1KBSplitter

About PVBUS4KBTo1KBSplitter

The purpose of this component is to allow an upstream component to access four downstream components in the same 4KB address range. It splits the 4KB range from 0 to 0xfff into the following four 1KB ranges, which allows four different components to be attached to the 4KB range:

- 0x0-0x3ff
- 0x400-0x7ff
- 0x800-0xbff
- 0xc00-0xfff

This overcomes a limitation of PVBUS which only allows components to be attached to memory addresses that are a multiple of 4KB in size.

Ports for PVBUS4KBTo1KBSplitter

Table 3-108: Ports

Name	Protocol	Type	Description
pvbus_m[4]	PVBUS	Master	The four downstream ports to be connected to peripherals. Each port covers 1KiB of the address space. Output address on each port will be in the range 0x0 - 0x03FF.
pvbus_s	PVBUS	Slave	The upstream port. Accepts addresses in range 0x0 - 0x0FFF. Outside of this range transactions will abort.

3.3.7 PVBUSCache

A PVBUSCache manages cache-line data and supports forwarding of transactions. This model is written in C++.

Ports for PVBUSCache

Table 3-109: Ports

Name	Protocol	Type	Description
bus_in[4]	PVBUS	Slave	Connections in from bus master.
bus_out[4]	PVBUS	Master	Connections out to bus slaves.
control	PVBUSCacheControl	Slave	Configuration/control port.
device	PVBUSCacheDevice	Master	Connection out to cache device.

3.3.8 PVBusDecoder

A PVBusDecoder allows bus transactions to be routed to one of many slaves, based on the address given in the transaction. This model is written in C++.

About PVBusDecoder

Each slave connection is associated with a specific address range on the `pvbus_m_range` port. In LISA+, the syntax for this is:

```
decoder.pvbus_m_range[start..end] = slave.pvbus
```

The values for `start` (inclusive) and `end` (inclusive) must specify a 4KB-aligned region of a multiple of 4K bytes. You can specify an address range for the slave, where the decoder remaps addresses into the appropriate range. The default address range for a slave is $[0-(\text{sizeofMasterRange} - 1)]$.

Ports for PVBusDecoder

Table 3-110: Ports

Name	Protocol	Type	Description
<code>pvbus_m_range</code>	PVBus	Master	Specifies the address range for the bus master. The range must be 4KB aligned and a multiple of 4KB in size. If the address range is larger than the size of the slave device, the slave is aliased.
<code>pvbus_s</code>	PVBus	Slave	Accepts incoming transactions. Connect this port to a bus master, or to the output of another bus decoder.

3.3.9 PVBusExclusiveMonitor

Global exclusive monitor. This model is written in C++.

Iris and MTI instances for PVBusExclusiveMonitor

This model has the following Iris instances:

Table 3-111: PVBusExclusiveMonitor Iris instances

InstanceName	ComponentName
<code>PVBusExclusiveMonitor</code>	<code>PVBusExclusiveMonitor</code>
<code>PVBusExclusiveMonitor.bus_mapper</code>	<code>PVBusMapper</code>

This model has the following MTI trace components:

Table 3-112: PVBusExclusiveMonitor MTI instances

InstanceName	ComponentName
<code>PVBusExclusiveMonitor</code>	<code>PVBusExclusiveMonitor</code>
<code>PVBusExclusiveMonitor.bus_mapper</code>	<code>PVBusMapper</code>

PVBusExclusiveMonitor contains the following CADI targets:

- `PVBusExclusiveMonitor`

Ports for PVBusExclusiveMonitor

Table 3-113: Ports

Name	Protocol	Type	Description
excl_cleared	Signal	Master	Exclusive monitor clear signal port.
pvbus_m	PVBus	Master	Bus master port.
pvbus_s	PVBus	Slave	Bus slave port.

Parameters for PVBusExclusiveMonitor

apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

log2_granule_size

log2 of address granule size

Type

int

Default value

0x0

match_access_width

Fail STREX if not the same access width as LDREX

Type

bool

Default value

0x0

match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

number_of_monitors

Number of monitors

Type

int

Default value

0x8

shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

3.3.10 PVBusExclusiveSquasher

Squashes the exclusive attribute on bus transactions. This model is written in LISA+.

Iris and MTI instances for PVBusExclusiveSquasher

This model has the following Iris instances:

Table 3-114: PVBusExclusiveSquasher Iris instances

InstanceName	ComponentName
PVBusExclusiveSquasher	PVBusExclusiveSquasher
PVBusExclusiveSquasher.bus_modifier	PVBusMapper

This model has the following MTI trace components:

Table 3-115: PVBusExclusiveSquasher MTI instances

InstanceName	ComponentName
PVBusExclusiveSquasher.bus_modifier	PVBusMapper

PVBusExclusiveSquasher contains the following CADI targets:

- PVBusExclusiveSquasher

Ports for PVBusExclusiveSquasher

Table 3-116: Ports

Name	Protocol	Type	Description
pvbus_m	PVBus	Master	Bus master port.
pvbus_s	PVBus	Slave	Bus slave port.

3.3.11 PVBusLogger

A PVBusLogger has a slave and a master port and traffic is passed straight through. All traffic is logged using an MTI trace event. This model is written in C++.

Iris and MTI instances for PVBusLogger

This model has the following Iris instances:

Table 3-117: PVBusLogger Iris instances

InstanceName	ComponentName
PVBusLogger	PVBusLogger
PVBusLogger.mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-118: PVBusLogger MTI instances

InstanceName	ComponentName
PVBusLogger	PVBusLogger
PVBusLogger.mapper	PVBusMapper

PVBusLogger contains the following CADI targets:

- PVBusLogger

Ports for PVBusLogger

Table 3-119: Ports

Name	Protocol	Type	Description
pvbus_m	PVBus	Master	Bus master port.
pvbus_s	PVBus	Slave	Bus slave port.

Parameters for PVBusLogger

trace_debug

Enable tracing of debug transactions

Type

bool

Default value

0x0

trace_snoops

Enable tracing of ACE snoop requests

Type

bool

Default value

0x0

3.3.12 PVBusMapper

Allow transactions to be remapped arbitrarily. This model is written in C++.

Iris and MTI instances for PVBusMapper

This model has the following Iris instances:

Table 3-120: PVBusMapper Iris instances

InstanceName	ComponentName
PVBusMapper	PVBusMapper

This model has the following MTI trace components:

Table 3-121: PVBusMapper MTI instances

InstanceName	ComponentName
PVBusMapper	PVBusMapper

About PVBusMapper

This component performs like `PVBusModifier`, but in addition:

- It has multiple downstream ports
- It allows routing of transactions to any one of these ports
- It allows arbitrary remapping of transaction addresses and attributes

As a generic modeling component, it does not have a hardware revision code.

For an example of how to use `PVBusMapper`, SEE `$PVLIB_HOME/examples/LISAPlus/RemappingWithPVBusMapper/`.

Ports for PVBusMapper

Table 3-122: Ports

Name	Protocol	Type	Description
control	PVBusMapperControl	Master	Configuration port to determine mappings.
pdbus_m[64]	PVBus	Master	Bus master ports.
pdbus_s	PVBus	Slave	Bus slave port.
reset	Signal	Slave	Reset signal.

3.3.13 PVBusMaster

The `PVBusMaster` subcomponent allows a device to generate `PVBus` transactions. This model is written in C++.

Iris and MTI instances for PVBusMaster

This model has the following Iris instances:

Table 3-123: PVBusMaster Iris instances

InstanceName	ComponentName
PVBusMaster	PVBusMaster

This model has the following MTI trace components:

Table 3-124: PVBusMaster MTI instances

InstanceName	ComponentName
PVBusMaster	PVBusMaster

About PVBusMaster

If you want a component to act as a bus master, instantiate a `PVBusMaster` subcomponent and then use its control port to create one or more transaction generators to generate transactions on the bus. `$PVLIB_HOME/examples/LISA/BusComponents/DmaTransfer.lisa` is an example component that shows this.

See also [1.13.2 TransactionGenerator efficiency considerations](#) on page 57.

Ports for PVBusMaster

Table 3-125: Ports

Name	Protocol	Type	Description
control	PVTransactionMaster	Slave	Enables the owning component to instantiate <code>pv::TransactionGenerator</code> objects.
pvbus_m	PVBus	Master	Sends out generated transactions to the bus.
reset	Signal	Slave	On the de-assert of this signal, a reset of the bus master will be latched this is used by the bus deadlock detection logic.

3.3.14 PVBusModifier

Allow transactions to be remapped arbitrarily. This model is written in C++.

Iris and MTI instances for PVBusModifier

This model has the following Iris instances:

Table 3-126: PVBusModifier Iris instances

InstanceName	ComponentName
PVBusModifier	PVBusMapper

This model has the following MTI trace components:

Table 3-127: PVBusModifier MTI instances

InstanceName	ComponentName
PVBusModifier	PVBusMapper



PVBusModifierx2 is an identical component, except it has two downstream ports.

Ports for PVBusModifier

Table 3-128: Ports

Name	Protocol	Type	Description
control	PVBusMapperControl	Master	Configuration port to determine mappings.
pvbus_m	PVBus	Master	Bus master port.
pvbus_s	PVBus	Slave	Bus slave port.
reset	Signal	Slave	Reset signal.

3.3.15 PVBusRouter

Allow transactions to be routed arbitrarily. This model is written in LISA+.

Iris and MTI instances for PVBusRouter

This model has the following Iris instances:

Table 3-129: PVBusRouter Iris instances

InstanceName	ComponentName
PVBusRouter	PVBusRouter
PVBusRouter.mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-130: PVBusRouter MTI instances

InstanceName	ComponentName
PVBusRouter.mapper	PVBusMapper

PVBusRouter contains the following CADI targets:

- PVBusRouter

Ports for PVBusRouter

Table 3-131: Ports

Name	Protocol	Type	Description
control	PVBusRouterControl	Master	Configuration port to determine filters.
pvbus_m[64]	PVBus	Master	Bus master ports.
pvbus_s	PVBus	Slave	Bus slave port.

3.3.16 PVBusSlave

A PVBusSlave handles incoming transactions, and handles support for mapping regions of device address space to work as RAM/ROM/device memory. This model is written in C++.

Iris and MTI instances for PVBusSlave

This model has the following Iris instances:

Table 3-132: PVBusSlave Iris instances

InstanceName	ComponentName
PVBusSlave	PVBusSlave

This model has the following MTI trace components:

Table 3-133: PVBusSlave MTI instances

InstanceName	ComponentName
PVBusSlave	PVBusSlave

About PVBusSlave

Any component that acts as a bus slave must:

- Provide a PVBus slave port.
- Instantiate a PVBusSlave subcomponent, with the size parameter configured for the address range covered by the device.
- Connect the slave port to the `pvbus_s` port on the PVBusSlave.

By default, the PVBusSlave translates all transactions into `read()` and `write()` requests on the device port.

The control port enables you to configure the PVBusSlave for a device. This lets you define the behavior of the different regions of the address space on the device. You can configure regions separately for read and write, at a 4KB granularity. This permits you to set memory-like, device-like, abort, or ignore access address regions.

Transactions to memory-like regions are handled internally by the PVBusSlave subcomponent. Abort and ignore regions are also handled by the PVBusSlave.

Transactions to device-like regions are forwarded to the device port. Your device must implement the `read()` and `write()` methods on the device port if any regions are configured as device-like.

This component typically does not significantly affect the performance of a PV system. However, correct implementation of the PVBusSlave component is critical to the performance of the overall PV system. For example, routing a request to a PVDevice port is slower than letting the PVBusSlave component handle the request internally. Arm recommends using the internal support for memory-like regions where possible.

Ports for PVBusSlave

Table 3-134: Ports

Name	Protocol	Type	Description
control	PVBusSlaveControl	Slave	Enables the owning component to control which regions of the device memory are to be handled as RAM/ROM/Device. These settings can be changed dynamically. For example, when a Flash component is being programmed, it can switch to treating reads as Device requests instead of ROM requests.
device	PVDevice	Master	Passes on requests for peripheral register accesses to permit the owning component to handle the request.
pvbus_s	PVBus	Slave	Handles incoming requests from bus masters.
reset	Signal	Slave	On the assert of this signal, a reset of the bus slave will be latched this is used by the bus deadlock detection logic.

3.3.17 PVMemoryProtectionEngine

Encrypt memory transactions for each encryption context with an independent key to prevent mismatch access. This model is written in C++.

Iris and MTI instances for PVMemoryProtectionEngine

This model has the following Iris instances:

Table 3-135: PVMemoryProtectionEngine Iris instances

InstanceName	ComponentName
PVMemoryProtectionEngine	PVMemoryProtectionEngine
PVMemoryProtectionEngine.mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-136: PVMemoryProtectionEngine MTI instances

InstanceName	ComponentName
PVMemoryProtectionEngine	PVMemoryProtectionEngine
PVMemoryProtectionEngine.mapper	PVBusMapper

PVMemoryProtectionEngine contains the following CADI targets:

- [PVMemoryProtectionEngine](#)

About PVMemoryProtectionEngine

PVMemoryProtectionEngine is a simplified implementation of a Memory Protection Engine (MPE) component as described in [Arm Realm Management Extension \(RME\) System Architecture](#).

PVMemoryProtectionEngine supports the following features:

- Memory encryption.

- Each 4KiB page in memory is encrypted based on an encryption key. Each Physical Address Space (PAS) has a separate encryption key.
- Two or more encryption keys can be the same value.
- Configurable encryption keys for each PAS.
- Configurable encryption block size.
- Configurable corruption strategy. You can control the behavior of memory contents that are not written by the access within the encryption block.
- Encryption/decryption algorithm is a simple XOR of data with the corresponding encryption key.
- Downstream memory is always stored as plain text, allowing debuggers to view data.

For example, if a block is currently encrypted by the ns-PAS and then a byte is written by the rl-PAS, if the `block_size_in_bytes` is 4KiB, the rest of the data in the 4KiB page is corrupted such that even if you read a different byte back through the ns-PAS, you would not get the original data.

The primary use case for this component is to identify software mis-programming, where the same Physical address is accessed through more than one PAS. With PVMemoryProtectionEngine enabled, a PE sees encrypted or corrupted data when it is accessed using a different PAS to the original PAS that wrote to that page in memory.

The PVMemoryProtectionEngine component is expected to be connected in a platform at the Point of Physical Aliasing (PoPA) if storage is shared, otherwise before each specific storage for a subset of the PASEs.

PVMemoryProtectionEngine imposes a runtime cost when enabled. Normally, it is only needed when debugging and verifying the Realm Management Monitor (RMM) software. If the RMM software is correct, memory contents encrypted with the wrong key would not be visible.

The PVMemoryProtectionEngine does not encrypt or corrupt the tag data for MTE, but this feature will be supported in future.

Ports for PVMemoryProtectionEngine

Table 3-137: Ports

Name	Protocol	Type	Description
<code>pvbus_m</code>	PVBus	Master	Manager ports of the MPE
<code>pvbus_s</code>	PVBus	Slave	Subordinate port of the MPE

Parameters for PVMemoryProtectionEngine

`block_size_in_bytes`

Encryption block size, supported sizes are 1 or 4096.

Type

int

Default value

0x1000

corruption_strategy

Corruption strategy (0 - fill with constants per-old-encryption-context, 1 - fill with constants per-new-encryption-context, 2 - random data).

Type

int

Default value

0x0

enable

Enables the Memory Protection Engine.

Type

bool

Default value

0x0

ignore_mecid

Ignore MECID during encryption key calculation

Type

bool

Default value

0x0

non_secure_pas_enc_key

Non-secure PAS encryption key.

Type

int

Default value

0x22

output_attributes_parameter_of_core

Encoding of various attributes on the bus.

Type

string

Default value

ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID,
ExtendedID[38]=MPAM_NS

realm_pas_enc_key

Realm PAS encryption key.

Type

int

Default value

0x88

root_pas_enc_key

Root PAS encryption key.

Type

int

Default value

0x44

secure_pas_enc_key

Secure PAS encryption key.

Type

int

Default value

0x11

3.3.18 PVWriteBuffer

The PVWriteBuffer subcomponent buffers PVBus transactions. This model is written in C++.

Iris and MTI instances for PVWriteBuffer

This model has the following Iris instances:

Table 3-138: PVWriteBuffer Iris instances

InstanceName	ComponentName
PVWriteBuffer	PVWriteBuffer
PVWriteBuffer.mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-139: PVWriteBuffer MTI instances

InstanceName	ComponentName
PVWriteBuffer	PVWriteBuffer
PVWriteBuffer.mapper	PVBusMapper

Ports for PVWriteBuffer

Table 3-140: Ports

Name	Protocol	Type	Description
barrier_notify_s	PVWriteBuffer_BarrierPort	Slave	Barrier notification input.
clk_in	ClockSignal	Slave	Clock input.
pvbus_m	PVBus	Master	Master connection to memory bus.
pvbus_s	PVBus	Slave	Slave connection for transactions to be buffered.
reset_in	Signal	Slave	Reset input.
serror_notify_m	PVWriteBuffer_SErrorPort	Master	SError output generation.

3.3.19 SimplePVBusMaster

Component to generate PVTransactions with configurable attributes and address. This model is written in LISA+.

Iris and MTI instances for SimplePVBusMaster

This model has the following Iris instances:

Table 3-141: SimplePVBusMaster Iris instances

InstanceName	ComponentName
SimplePVBusMaster	SimplePVBusMaster
SimplePVBusMaster.clocktimer64	ClockTimerThread64
SimplePVBusMaster.clocktimer64.thread	SchedulerThread
SimplePVBusMaster.clocktimer64.thread_event	SchedulerThreadEvent
SimplePVBusMaster.pvbusmaster	PVBusMaster
SimplePVBusMaster.pvbusslave	PVBusSlave

This model has the following MTI trace components:

Table 3-142: SimplePVBusMaster MTI instances

InstanceName	ComponentName
SimplePVBusMaster.pvbusmaster	PVBusMaster
SimplePVBusMaster.pvbusslave	PVBusSlave

SimplePVBusMaster contains the following CADI targets:

- ClockTimerThread64
- SchedulerThread
- SchedulerThreadEvent
- SimplePVBusMaster

Ports for SimplePVBUSMaster

Table 3-143: Ports

Name	Protocol	Type	Description
pvbust_m	PVBus	Master	Output of generated transactions.
pvbust_s	PVBus	Slave	-

Parameters for SimplePVBUSMaster

verbose

verbose

Type

bool

Default value

0x0

3.3.20 TZSwitch

Allow TrustZone secure/normal bus signals to be routed separately. This model is written in LISA+.

Iris and MTI instances for TZSwitch

This model has the following Iris instances:

Table 3-144: TZSwitch Iris instances

InstanceName	ComponentName
TZSwitch	TZSwitch
TZSwitch.pvbust_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-145: TZSwitch MTI instances

InstanceName	ComponentName
TZSwitch.pvbust_mapper	PVBusMapper

TZSwitch contains the following CADI targets:

- TZSwitch

About TZSwitch

The TZSwitch component allows you to control transaction routing based on the TrustZone® security state of the transaction.

The default behavior is to forward secure transactions to `pvbust_port_a`, and normal transactions to `pvbust_port_b`.

You must only use TZSwitches if the routing decisions change infrequently, for example as part of a memory remap.

The `secure` and `normal` parameters control the initial state of this component:

- 0** Ignore these transactions.
- 1** Forward the transactions to `pvbus_port_a`.
- 2** Forward the transactions to `pvbus_port_b`.
- 3** Generate an abort for these transactions.

The numbers used for initial configuration are not the same as the enumeration constants used to control routing at runtime.

Ports for TZSwitch

Table 3-146: Ports

Name	Protocol	Type	Description
<code>control</code>	<code>TZSwitchControl</code>	Slave	Controls routing of transactions.
<code>pvbus_input</code>	PVBus	Slave	Slave port for connection to PVBus master/decoder.
<code>pvbus_port_a</code>	PVBus	Master	Output port a.
<code>pvbus_port_b</code>	PVBus	Master	Output port b.

Parameters for TZSwitch

normal

Normal Port

Type

int

Default value

0x2

secure

Secure Port

Type

int

Default value

0x1

3.4 Clocking components

This section describes the Clocking components.

The clocking components and protocols provide a mechanism for systems to regulate the execution rate of components. Clocking includes the concept of clock rates, dividers to change clock rates, and timers to generate callbacks based on those clock rates.

If the MasterClock component is instantiated in a system, it provides a consistent master clock rate. Although this rate is not defined, you can consider this to be 1Hz, even for non-SystemC systems. ClockDivider components are able to convert this clock rate into a new rate using a multiplier and divider, although the clock rate cannot be divided to be less than 1Hz. You can cascade ClockDivider components to produce many different clock rates within a system. The maximum ratio of any two clocks in the system must be less than 2^{32} .

ClockTimer components can be instantiated by a component and connected to any MasterClock or ClockDivider output. ClockTimers can generate callbacks after a given number of ticks of that clock. ClockTimers can invoke a behavior on the component to permit the component to perform work. The component can then request the ClockTimer to repeat its count.

3.4.1 ClockDivider

A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters. This model is written in C++.

Iris and MTI instances for ClockDivider

This model has the following Iris instances:

Table 3-147: ClockDivider Iris instances

InstanceName	ComponentName
ClockDivider	ClockDivider

This model has the following MTI trace components:

Table 3-148: ClockDivider MTI instances

InstanceName	ComponentName
ClockDivider	ClockDivider

ClockDivider contains the following CADI targets:

- ClockDivider

About ClockDivider

This component uses a configurable ratio to convert the ClockSignal rate at its input to a new ClockSignal rate at its output. Changes to the input rate or ratio take effect immediately and clocking components dependent on the output rate continue counting at the new rate.

This component does not normally incur a runtime performance cost. However, reprogramming the clock rate causes all related clocks and timers to be recalculated.

For an example of the use of ClockDividers, see the `vEMotherBoard.lisa` component in the `$PVLIB_HOME/examples/LISA/FVP_VE/LISA/` directory of your Fast Models distribution.

ClockDivider parameters

The ClockDivider parameters are not exposed by CADI. You can set them from LISA but not from SystemC:

- mul

Clock rate multiplier.
Type: `uint64_t`. Default value: 1.
- div

Clock rate divider.
Type: `uint64_t`. Default value: 1.

Ports for ClockDivider

Table 3-149: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Input clock signal, coming from a MasterClock or another ClockDivider.
clk_out	ClockSignal	Master	Clock signal generated by this ClockDivider.
rate	ClockRateControl	Slave	Permits you to dynamically change the clock divider ratio.

Parameters for ClockDivider

- div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1
- mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

3.4.2 ClockGate_cpp

Clock gate for dis/enabling the clock. This model is written in C++.

Iris and MTI instances for ClockGate_cpp

This model has the following Iris instances:

Table 3-150: ClockGate_cpp Iris instances

InstanceName	ComponentName
ClockGate_cpp	ClockGate
ClockGate_cpp.ClockDivider	ClockDivider

This model has the following MTI trace components:

Table 3-151: ClockGate_cpp MTI instances

InstanceName	ComponentName
ClockGate_cpp.ClockDivider	ClockDivider

ClockGate_cpp contains the following CADI targets:

- ClockGate

Ports for ClockGate_cpp

Table 3-152: Ports

Name	Protocol	Type	Description
clk_enable	Signal	Slave	-
clk_in	ClockSignal	Slave	-
clk_out	ClockSignal	Master	-
halt	Signal	Master	-

Parameters for ClockGate_cpp

diagnostics

Diagnostics

Type

int

Default value

0x0

divider.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

divider.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

3.4.3 ClockSelector_cpp

ClockSignal Selector. This model is written in C++.

Iris and MTI instances for ClockSelector_cpp

This model has the following Iris instances:

Table 3-153: ClockSelector_cpp Iris instances

InstanceName	ComponentName
ClockSelector_cpp	ClockSelector
ClockSelector_cpp.ClockDivider	ClockDivider

This model has the following MTI trace components:

Table 3-154: ClockSelector_cpp MTI instances

InstanceName	ComponentName
ClockSelector_cpp.ClockDivider	ClockDivider

ClockSelector_cpp contains the following CADI targets:

- ClockSelector

Ports for ClockSelector_cpp

Table 3-155: Ports

Name	Protocol	Type	Description
clk_in[11]	ClockSignal	Slave	-
clk_out	ClockSignal	Master	-
clk_sel	Signal	Slave	-
clk_sel_num	Value	Slave	-

Parameters for ClockSelector_cpp

clkdiv0.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv0.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv1.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv1.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv10.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv10.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv2.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv2.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv3.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv3.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv4.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv4.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv5.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv5.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv6.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv6.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv7.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv7.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv8.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv8.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv9.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv9.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdivider.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdivider.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

diagnostics

Diagnostics

Type

int

Default value

0x0

3.4.4 ClockTimer

A ClockTimer provides support for counting a number of ticks at the rate determined by the input clock. When the countdown reaches zero, the timer can send a signal to another component. That component can return a value that will cause the timer to start counting down again. Setting up a timer is very efficient, and no host processing time is used while a counter is counting down: when a timer is started, the scheduler precomputes the finish time. To use a ClockTimer connect the ClockTimer's 'timer_callback' port to a slave port that implements the 'signal()' behaviour and connect a clock signal to the clk_in port. Use the 'timer_control' port to start the timer counting down for a given number of ticks. This model is written in C++.

About ClockTimer

This component provides a mechanism for other components to schedule a callback after a number of ticks at a given ClockSignal rate.

An active ClockTimer component incurs no simulation overhead. For best performance, avoid frequently canceling timers or querying the number of remaining ticks in your performance-critical code.

Ports for ClockTimer

Table 3-156: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Determines the tick rate of the timer.
timer_callback	TimerCallback	Master	Port on which a signal is sent after the number of scheduled ticks has elapsed.
timer_control	TimerControl	Slave	Permits the timer to be set, canceled and queried.

3.4.5 ClockTimer64

A ClockTimer64 provides support for counting a number of ticks at the rate determined by the input clock. When the countdown reaches zero, the timer can send a signal to another component. That component can return a value that will cause the timer to start counting down again. Setting up a timer is very efficient, and no host processing time is used while a counter is counting down:

when a timer is started, the scheduler precomputes the finish time. This version of the timer provides 64 bit resolution. This model is written in C++.

About ClockTimer64

This component provides a mechanism for other components to schedule a callback after a number of ticks at a given ClockSignal rate.

An active ClockTimer64 component incurs no simulation overhead. For best performance, avoid having your performance-critical code frequently cancel timers or query the number of remaining ticks.

Ports for ClockTimer64

Table 3-157: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Determines the tick rate of the timer.
timer_callback	TimerCallback64	Master	Port on which a signal is sent after the number of scheduled ticks has elapsed.
timer_control	TimerControl64	Slave	Permits the timer to be set, canceled and queried.

3.4.6 ClockTimerThread

A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This means that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64). This model is written in LISA+.

Iris and MTI instances for ClockTimerThread

This model has the following Iris instances:

Table 3-158: ClockTimerThread Iris instances

InstanceName	ComponentName
ClockTimerThread	ClockTimerThread
ClockTimerThread.timer	ClockTimerThread64
ClockTimerThread.timer.thread	SchedulerThread
ClockTimerThread.timer.thread_event	SchedulerThreadEvent

ClockTimerThread contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- SchedulerThread
- SchedulerThreadEvent

Ports for ClockTimerThread

Table 3-159: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Determines the tick rate of the timer.
timer_callback	TimerCallback	Master	Port on which a signal is sent after the number of scheduled ticks has elapsed.
timer_control	TimerControl	Slave	Permits the timer to be set, canceled and queried.

3.4.7 ClockTimerThread64

A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64). This model is written in LISA+.

Iris and MTI instances for ClockTimerThread64

This model has the following Iris instances:

Table 3-160: ClockTimerThread64 Iris instances

InstanceName	ComponentName
ClockTimerThread64	ClockTimerThread64
ClockTimerThread64.thread	SchedulerThread
ClockTimerThread64.thread_event	SchedulerThreadEvent

ClockTimerThread64 contains the following CADI targets:

- ClockTimerThread64
- SchedulerThread
- SchedulerThreadEvent

Ports for ClockTimerThread64

Table 3-161: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Determines the tick rate of the timer.
timer_callback	TimerCallback64	Master	Port on which a signal is sent after the number of scheduled ticks has elapsed.
timer_control	TimerControl64	Slave	Permits the timer to be set, canceled and queried.

3.4.8 MasterClock

A source of a clock signal representing the base clock rate of the simulation (nominally 1Hz). The signal from the output port can be connected to a ClockDivider, to generate clock signals

at a different clock rate. The output can also be connected to a ClockTimer in order to generate scheduled events, or to any other components that accept a ClockSignal input. See ClockSignal.lisa for more information. This model is written in C++.

About MasterClock

This component provides a single ClockSignal output that can be used to drive the ClockSignal input of ClockDividers, ClockTimers and other clocking components.

The rate of the MasterClock is not defined because all clocking is relative, but can be considered to be 1 Hz.

A system might contain more than one MasterClock, all of which generate the same ClockSignal rate.

Ports for MasterClock

Table 3-162: Ports

Name	Protocol	Type	Description
clk_out	ClockSignal	Master	Master clock rate.

3.4.9 PLLControl_cpp

Simulate PLL clock frequency control logic. This model is written in C++.

Iris and MTI instances for PLLControl_cpp

This model has the following Iris instances:

Table 3-163: PLLControl_cpp Iris instances

InstanceName	ComponentName
PLLControl_cpp	PLLControl
PLLControl_cpp.ClockDivider	ClockDivider

This model has the following MTI trace components:

Table 3-164: PLLControl_cpp MTI instances

InstanceName	ComponentName
PLLControl_cpp.ClockDivider	ClockDivider

PLLControl_cpp contains the following CADI targets:

- PLLControl

Ports for PLLControl_cpp

Table 3-165: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	-
clk_out	ClockSignal	Master	-
lock	Signal	Master	-
rate	ClockRateControl	Slave	-
unlock	Signal	Master	-

Parameters for PLLControl_cpp

clkdiv.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

3.4.10 ScalableClockControl_cpp

Clock control allows input selection, rate control and gating. This model is written in C++.

Iris and MTI instances for ScalableClockControl_cpp

This model has the following Iris instances:

Table 3-166: ScalableClockControl_cpp Iris instances

InstanceName	ComponentName
ScalableClockControl_cpp	ScalableClockControl
ScalableClockControl_cpp.ClockDivider	ClockDivider
ScalableClockControl_cpp.ClockGate	ClockGate
ScalableClockControl_cpp.ClockGate.ClockDivider	ClockDivider
ScalableClockControl_cpp.ClockSelector	ClockSelector
ScalableClockControl_cpp.ClockSelector.ClockDivider	ClockDivider

This model has the following MTI trace components:

Table 3-167: ScalableClockControl_cpp MTI instances

InstanceName	ComponentName
ScalableClockControl_cpp.ClockDivider	ClockDivider
ScalableClockControl_cpp.ClockGate.ClockDivider	ClockDivider
ScalableClockControl_cpp.ClockSelector.ClockDivider	ClockDivider

ScalableClockControl_cpp contains the following CADI targets:

- ScalableClockControl

Ports for ScalableClockControl_cpp

Table 3-168: Ports

Name	Protocol	Type	Description
clk_out	ClockSignal	Master	-
clkEnable	Signal	Slave	-
clkSel	Value	Slave	-
clock_in[10]	ClockSignal	Slave	-
clock_rate[10]	ClockRateControl	Slave	-
halt	Signal	Master	-
refClk_in	ClockSignal	Slave	-

Parameters for ScalableClockControl_cpp

clkDiv1.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv1.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv10.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv10.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv2.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv2.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv3.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv3.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv4.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv4.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv5.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv5.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv6.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv6.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv7.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv7.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv8.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv8.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv9.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv9.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkGate.diagnostics

Diagnostics

Type

int

Default value

0x0

clkGate.divider.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkGate.divider.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv0.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv0.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv1.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv1.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv10.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv10.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv2.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv2.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv3.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv3.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv4.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv4.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv5.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv5.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv6.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv6.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv7.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv7.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv8.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv8.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv9.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv9.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type
int

Default value
0x1

clkSelect.clkdivider.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type
int

Default value
0x1

clkSelect.clkdivider.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type
int

Default value
0x1

clkSelect.diagnostics

Diagnostics

Type
int

Default value
0x0

3.4.11 SwitchedClockControl_cpp

Clock control allows input selection, rate control and gating. This model is written in C++.

Iris and MTI instances for SwitchedClockControl_cpp

This model has the following Iris instances:

Table 3-169: SwitchedClockControl_cpp Iris instances

InstanceName	ComponentName
SwitchedClockControl_cpp	SwitchedClockControl
SwitchedClockControl_cpp.ClockDivider	ClockDivider
SwitchedClockControl_cpp.ClockGate	ClockGate
SwitchedClockControl_cpp.ClockGate.ClockDivider	ClockDivider

InstanceName	ComponentName
SwitchedClockControl_cpp.ClockSelector	ClockSelector
SwitchedClockControl_cpp.ClockSelector.ClockDivider	ClockDivider

This model has the following MTI trace components:

Table 3-170: SwitchedClockControl_cpp MTI instances

InstanceName	ComponentName
SwitchedClockControl_cpp.ClockDivider	ClockDivider
SwitchedClockControl_cpp.ClockGate.ClockDivider	ClockDivider
SwitchedClockControl_cpp.ClockSelector.ClockDivider	ClockDivider

SwitchedClockControl_cpp contains the following CADI targets:

- SwitchedClockControl

Ports for SwitchedClockControl_cpp

Table 3-171: Ports

Name	Protocol	Type	Description
clk_out	ClockSignal	Master	-
clkDivExt	ClockRateControl	Slave	-
clkDivSys	ClockRateControl	Slave	-
clkEnable	Signal	Slave	-
clkSel	Value	Slave	-
halt	Signal	Master	-
refClk_in	ClockSignal	Slave	-
sysClk_in	ClockSignal	Slave	-
xClk_in	ClockSignal	Slave	-

Parameters for SwitchedClockControl_cpp

clkDiv1.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv1.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv2.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkDiv2.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkGate.diagnostics

Diagnostics

Type

int

Default value

0x0

clkGate.divider.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkGate.divider.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv0.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv0.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv1.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv1.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv10.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv10.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv2.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv2.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv3.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv3.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv4.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv4.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv5.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv5.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv6.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv6.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv7.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv7.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv8.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv8.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv9.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdiv9.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdivider.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.clkdivider.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkSelect.diagnostics

Diagnostics

Type

int

Default value

0x0

3.5 Core components

This section describes the Core components.

Code Translation (CT) components translate instructions on the fly and cache the translations to enable fast execution of code, but sacrificing timing accuracy. They also use efficient PV bus models to enable fast access to memory and devices.

CT components differ in some ways from the modeled IP:

- They do not model cycle timing. In aggregate, all instructions execute in one processor master clock cycle, except for Wait For Interrupt.
- Write buffers are not modeled on all processors.
- Most aspects of TLB behavior are implemented in the models. In Arm®v7 models and later, the TLB memory attribute settings are used when stateful cache is enabled.
- No device-accurate MicroTLB is implemented.
- Device-accurate modeling of multiple TLBs is off by default.
- A single memory access port is implemented. The port combines accesses for instruction, data, DMA, and peripherals. Configuration of the peripheral port memory map register is ignored.
- All memory accesses are atomic and are performed in Programmer's View order. Unaligned accesses are always performed as byte transfers.
- Some instruction sequences are executed atomically so that system time does not advance during their execution. This difference in behavior can affect sequential accesses of device registers where devices are expecting time to move on between each access.
- Interrupts are not taken at every instruction boundary.
- Integration and test registers are not implemented.
- Models do not support running Software Test Libraries (STLs).
- Not all CP14 debug registers are implemented on all processors.
- Breakpoint types that the models support directly are:
 - Single address unconditional instruction breakpoints.
 - Single address unconditional data breakpoints.
 - Unconditional instruction address range breakpoints.

- Pseudoregisters in the debugger support processor exception breakpoints. Setting an exception register to a nonzero value stops execution on entry to the associated exception vector.
- Cluster models do not simulate all cores at the same time. They execute a number of instructions on each core in turn. There can be a bias in the order in which cores run after a restart (for example, core 0 always runs first), so the simulation might hit breakpoints on the favored core more often.
- Performance counters are not implemented on all models.
- Some models implement caches, although all processor models implement cache control registers.
- ECC and parity schemes are hardware-specific so are not modeled.
- Models use a simplified view of the external buses.
- Clusters based on DSU-110 and later support OFF, ON, AND DBG_RECOV modes.

3.5.1 AEMv8RMPCT

AEMv8RMPCT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-172: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `has_restriction_on_speculative_data_loaded`

Parameters removed:

- `cpu0.CONFIG64`
- `cpu0.POWERCTLI`
- `cpu1.POWERCTLI`
- `cpu2.POWERCTLI`
- `cpu3.POWERCTLI`
- `has_restrictions_on_speculative_data_loaded`
- `max_32bit_el`

Iris and MTI instances for AEMv8RMPCT

This model has the following Iris instances:

Table 3-173: AEMv8RMPCT Iris instances

InstanceName	ComponentName
AEMv8RMPCT	Cluster_ARMAEMv8-R_MP
AEMv8RMPCT.AMU	PVBusLogger
AEMv8RMPCT.AMU.mapper	PVBusMapper
AEMv8RMPCT.DAP	PVBusLogger
AEMv8RMPCT.DAP.mapper	PVBusMapper
AEMv8RMPCT.MMAP	PVBusLogger
AEMv8RMPCT.MMAP.mapper	PVBusMapper
AEMv8RMPCT.acp_mapper	PVBusMapper
AEMv8RMPCT.cpu0	ARMAEMv8-R_MP
AEMv8RMPCT.cpu0.UTLB	TLB
AEMv8RMPCT.cpu0.dtlb	TlbCadi
AEMv8RMPCT.cpu0.l1dcache	PVCache
AEMv8RMPCT.cpu0.l1dcache.upstream[0]	PVBusSlave
AEMv8RMPCT.cpu0.l1icache	PVCache
AEMv8RMPCT.cpu0.l1icache.upstream[0]	PVBusSlave
AEMv8RMPCT.ext_bus	PVBusLogger
AEMv8RMPCT.ext_bus.mapper	PVBusMapper
AEMv8RMPCT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
AEMv8RMPCT.l2_cache	PVCache
AEMv8RMPCT.l2_cache.upstream[0]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[10]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[11]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[12]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[13]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[14]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[15]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[16]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[1]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[2]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[3]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[4]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[5]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[6]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[7]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[8]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[9]	PVBusSlave
AEMv8RMPCT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-174: AEMv8RMPCT MTI instances

InstanceName	ComponentName
AEMv8RMPCT	ARMv8Cluster
AEMv8RMPCT.AMU	PVBusLogger
AEMv8RMPCT.AMU.mapper	PVBusMapper
AEMv8RMPCT.DAP	PVBusLogger
AEMv8RMPCT.DAP.mapper	PVBusMapper
AEMv8RMPCT.MMAP	PVBusLogger
AEMv8RMPCT.MMAP.mapper	PVBusMapper
AEMv8RMPCT.acp_mapper	PVBusMapper
AEMv8RMPCT.cpu0	ARM_AEMv8-R_MP
AEMv8RMPCT.cpu0.UTLB	TLB
AEMv8RMPCT.cpu0.l1dcache	PVCache
AEMv8RMPCT.cpu0.l1dcache.upstream[0]	PVBusSlave
AEMv8RMPCT.cpu0.l1icache	PVCache
AEMv8RMPCT.cpu0.l1icache.upstream[0]	PVBusSlave
AEMv8RMPCT.ext_bus	PVBusLogger
AEMv8RMPCT.ext_bus.mapper	PVBusMapper
AEMv8RMPCT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
AEMv8RMPCT.l2_cache	PVCache
AEMv8RMPCT.l2_cache.upstream[0]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[10]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[11]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[12]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[13]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[14]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[15]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[16]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[1]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[2]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[3]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[4]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[5]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[6]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[7]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[8]	PVBusSlave
AEMv8RMPCT.l2_cache.upstream[9]	PVBusSlave
AEMv8RMPCT.l2_flusher	AsyncCacheFlushUnit

AEMv8RMPCT contains the following CADI targets:

- ARMAEMv8-R_MP
- Cluster_ARMAEMv8-R_MP
- PVCache
- TlbCadi

About AEMv8RMPCT

AEMv8RMPCT allows you to target AArch32 or AArch64, RAS, VFP, EL2, and other Arm®v8-R features.

This table lists the major differences between AEMv8RMPCT and Arm Cortex®-R82 Fast Models:

Table 3-175: Major differences between AEMv8RMPCT and Arm Cortex-R82 Fast Models

Feature	AEMv8R	Cortex-R82
VMSA_supported	Configurable	Not available
has_aarch64	Configurable	Not available
has_pl2	Configurable	Always true
has_pmu	Configurable	Always true
has_ras	Configurable	Always true
PA_SIZE	Configurable	Always 40
Armv8.5 feature-specific parameters	Configurable	Not available
stage1_tlb_size	Configurable	Not available
stage12_tlb_size	Configurable	Not available
has_writebuffer	Configurable	Not available
vfp-present	Configurable	Always true
def_mem_map	Configurable	Fixed, according to the specification
IMPDEF registers, for example all IMP_* registers in the R82 specification.	Not available	Supported

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (crypto.dll or crypto.so) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for AEMv8RMPCT

Table 3-176: Ports

Name	Protocol	Type	Description
broadcastcachemaint	Signal	Slave	Enable broadcasting of cache maintenance operations to downstream caches.
broadcastinner	Signal	Slave	Enable broadcasting of Inner Shareable transactions.
broadcastouter	Signal	Slave	Enable broadcasting of Outer Shareable transactions.
cache_validation_control	Value	Slave	This signal provides default exception handling state.

Name	Protocol	Type	Description
cfgend[4]	Signal	Slave	This signal is for EE bit initialisation
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
CNTHPIRQ[4]	Signal	Master	Timer signals to SOC
CNTHPSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTHVSIHQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC
CNTPSIHQ[4]	Signal	Master	Timer signals to SOC
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module
CNTVIRQ[4]	Signal	Master	Timer signals to SOC
commirq[4]	Signal	Master	Interrupt signal from debug communication channel.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers
cpuporeset[4]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	-
cti0extin[4]	Signal	Slave	-
cti0extout[4]	Signal	Master	-
cti1extin[4]	Signal	Slave	-
cti1extout[4]	Signal	Master	-
cti2extin[4]	Signal	Slave	-
cti2extout[4]	Signal	Master	-
cti3extin[4]	Signal	Slave	-
cti3extout[4]	Signal	Master	-
ctidbgirq[4]	Signal	Master	-
dbgen[4]	Signal	Slave	-
dbgnopwrdown[4]	Signal	Master	These signals relate to core power down.
dbgpwrdownack[4]	Signal	Master	Debug power down acknowledge.
dbgpwrdownreq[4]	Signal	Slave	Debug power down request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
ext_slave_s	PVBus	Slave	External Slave port. Equivalent to AXIS port.
external_trace_reset[4]	Signal	Slave	ETMv4 External Trace Reset signal.

Name	Protocol	Type	Description
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
flash_m[4]	PVBus	Master	Flash Port
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
hiden[4]	Signal	Slave	External debug interface.
hniden[4]	Signal	Slave	External debug interface.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
irqs[224]	Signal	Slave	These signals drive the CPU's interrupt controller interrupt lines.
l2reset	Signal	Slave	This signal resets timer and interrupt controller and l2cache
llpp_m[4]	PVBus	Master	LLPP (Low-Latency Peripheral Port).
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[4]	Signal	Slave	External debug interface.
periphbase	Value_64	Slave	This port sets the base address of private peripheral region
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg[4]	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt.
reset[4]	Signal	Slave	-
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
rvbar[4]	Value_64	Slave	Reset vector base address.
sei[4]	Signal	Slave	Per core System Error physical pins
smpnamp[4]	Signal	Master	This signals AMP or SMP mode for each core
spiden[4]	Signal	Slave	Secure invasive debug enable.
spniden[4]	Signal	Slave	Secure non-invasive debug enable.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state
teinit[4]	Signal	Slave	This signal provides default exception handling state.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trace_unit_reset[4]	Signal	Slave	ETMv4 Trace Unit Reset signal.

Name	Protocol	Type	Description
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for AEMv8RMPCT

ADFSR-AIFSR-implemented

ADFSR and AIFSR are implemented

Type

bool

Default value

0x0

AIDR

Value of AIDR_EL1 register.

Type

int

Default value

0x0

AMIIDR

Value of AMU Implementation Identification Register

Type

int

Default value

0x43b

AMPIDR

Value of AMU Peripheral Identification Register

Type

int

Default value

0x4000bb000

BPIMVA_causes_translation_lookup

Do a translation when BPIMVA instruction is executed (which may cause a translation fault).

Type

bool

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation to NC/DEV memory. The broadcastatomic signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTATOMICL

Enable broadcasting of atomic operation to LLRAM memory. The broadcastatomicl signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTCACHEMAIN

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CCSIDR-L1D_override

If nonzero, override the value presented in CCSIDR for L1D (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L1I_override

If nonzero, override the value presented in CCSIDR for L1I (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L2_override

If nonzero, override the value presented in CCSIDR for L2 (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L3_override

If nonzero, allow L3 selection in CSSELR and present this value in CCSIDR (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CFGTFPEN_pin_reset

CFGTFPEN Configuration pin at reset for bitfield IMP_MEMPROTCTLR_EL1.TFPEN.

Type

bool

Default value

0x0

CHI

Selects the type of protocol the Main Manager(MM) interface implements. 0, MM port configured as AXI. 1, MM port configured as CHI.

Type

bool

Default value

0x0

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

CTIPIDR

If non-zero, override the CTI Peripheral Identification Register

Type

int

Default value

0x0

CTR-L1Ip-override

If non-zero, override the L1Ip bits in CTR/CTR_ELO system register. This does not change the behaviour of the cache, only what is present in the CTR register.

Type

int

Default value

0x0

DBGBCR_BT_applies_RES0_before_valid_check

If true, RES0 behaviour is applied to DBGBCR(_EL1).BT before checking for reserved values for this field.

Type

bool

Default value

0x1

DBGPIDR

If non-zero, override the Debug Peripheral Identification Register

Type

int

Default value

0x0

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

ERRIIDR

Value of RAS Implementation Identification Register

Type

int

Default value

0xd800143b

ERRPIDR

Value of RAS Peripheral Identification Register

Type

int

Default value

0x4100bbd80

ERXMISC0_mask

Write Mask for ERXMISC0 RAS Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

MIDR

Value of MIDR_EL1 register.

Type

int

Default value

0x410fd0f0

NUM_CORES

Number of cores in cluster

Type

int

Default value

0x1

PA_SIZE

Physical address range supported (FEAT_LPA).

Type

int

Default value

0x28

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

PMCEID0

Performance Monitor Common Event ID Reg 0 value - 64 bit

Type

int

Default value

0xffffffff

PMCEID1

Performance Monitor Common Event ID Reg 1 value - 64 bit

Type

int

Default value

0xffffffff

PMSIDR.ArchInst

Defines whether architecture instruction sampling is implemented or not, if not only micro op sampling is implemented. Model only supports architecture instruction sampling, but allows ID register field to be configured.

Type

bool

Default value

0x1

PMSIDR.CRR

Defines whether call return branch records (FEAT_SPE_CRR) is implemented or not

Type

bool

Default value

0x0

PMSIDR.LDS

Defines whether data source for sampled load instruction is implemented or not. Model does not implement loaded data source, but allows ID register field to be configured.

Type

bool

Default value

0x0

PMUPIDR

If non-zero, override the PMU Peripheral Identification Register

Type

int

Default value

0x0

VMSA_supported

VMSA is supported at EL1

Type

bool

Default value

0x1

abort_execution_from_device_memory

Execution from device memory generates a prefetch abort.

Type

bool

Default value

0x0

advsimd_overread

AdvSIMD element load operations access all bytes of a 16-byte aligned window, even in Device memory

Type

bool

Default value

0x0

align_pc_on_branch_to_unaligned_pc_aarch32

Force PC align for branches to an unaligned PC counter in A32 state

Type

bool

Default value

0x0

align_pc_on_debug_exit_to_aarch32

Exit to AARCH32 state from debug state forces pc bit0 to 0

Type

bool

Default value

0x0

align_pc_on_illegal_exception_return_to_aarch32

Align PC when performing an illegal exception return from AArch64 to AArch32.

Type

bool

Default value

0x1

amu_aux_type_fixed

Lists which AMU auxiliary registers that are fixed and to which event type. The JSON schema is: {fixed_aux_reg:evt_type, ...}. For example {"0":0x300} would make auxiliary register 0 fixed to event type 0x300

Type

string

Default value**amu_mmap_address**

AMU base address for each core on system bus. 0 means the AMU is not mapped, otherwise the address must be 4KB aligned. JSON schema for the parameter value is: {"format":"all_addrs_are_absolute_wrt_systembus", "cores": [{"amu":0x0}, {"amu":0x0}, {"amu":0x0}, {"amu":0x0}]}

Type

string

Default value**amu_num_auxiliary_counters**

Number of AMU auxiliary counters implemented

Type

int

Default value

0x0

apshr_read_restrict

At EL0, unknown bits of APSR are RAZ.

Type

bool

Default value

0x0

atomic_memtype_fault_priority

This parameter describes the priority of unsupported atomic/exclusive memtype fault w.r.t alignment and permission fault. 0, BEFORE_ALIGN_MEM_FAULT. 1, AFTER_ALIGN_BEFORE_PERM_FAULT. 2, AFTER_PERM_FAULT.

Type

int

Default value

0x0

auxilliary_feature_register0

Value of AFR0 ID register.

Type

int

Default value

0x0

branch-predictor-clear-policy

Set branch prediction policy as defined for MMFR1[31:28]. This does not change the behaviour of the branch predictor, only what is reported in MMFR1.BPred.

Type

int

Default value

0x2

branch-predictor-supported-ops

Set branch prediction policy as defined for MMFR3[11:8]. This does not change the behaviour of the branch predictor, only what is reported in MMFR3.BPMaint.

Type

int

Default value

0x1

bus_protection_enable_at_reset

Enable TCM, L1Cache bus protection after reset

Type

bool

Default value

0x0

cache-log2linelen

Log2 of the cache line length in bytes.

Type

int

Default value

0x6

cache_maintenance_hits_watchpoints

DCIMVA operations executed in AArch32 modes hit watchpoints.

Type

bool

Default value

0x0

changing_block_size_without_bbm_support

Level of support for changing block size without break-before-make (FEAT_BBM).

Type

int

Default value

0x0

check_memory_attributes

Detect and report TLB use of conflicting memory attributes for views of the same physical address

Type

bool

Default value

0x1

clean_invalidate_cache_on_warm_reset

Clean and invalidate caches on warm reset

Type

bool

Default value

0x0

clear_reg_top_eret

Behaviour of the upper 32-bits of the Xn registers when changing between AArch32 state and AArch64 state. 0, upper 32-bits preserved for all registers. 1, upper 32-bits set to 0 for all accessible registers. 2, upper 32-bits set to 0 for a random selection of accessible registers. 3, upper-32-bits set to 0 for registers touched in AArch32

Type

int

Default value

0x1

clear_reg_top_set

Whether to clear upper 32-bits of the Xn register when corresponding AArch32 register is set via CADI/Iris.

Type

bool

Default value

0x1

cluster_utid

Unique cluster transaction identifier for interconnect protection

Type

int

Default value

0x0

configure_pmu_events_with_json

"Configure v8.6 and newer PMU events. Note : This param has high priority and overrides the setting of "has_*_pmu_events" if both the params are provided. JSON schema for the parameter value is e.g. 1. {"all":false} 2. {"pmu_events":["EVENT_NAME_1","EVENT_NAME_2"]}"

Type

string

Default value**configure_v8_6_pmu_events_with_json**

"[DEPRECATED: Set configure_pmu_events_with_json to the same value instead] Configure v8.6 PMU events. Note : This param has high priority and overrides the setting of "has_v8_6_pmu_events" if both the params are provided. JSON schema for the parameter value is e.g. 1. {"all":false} 2. {"pmu_events":["BR_INDNR_RETIRE", "BR_IND_RETIRE", "BR_RETURN_SKIP_RETIRE", "BR_RETURN_ANY_RETIRE", "BR_INDNR_SKIP_RETIRE",

"BR_INDNR_TAKEN_RETIRED", "BR_IND_SKIP_RETIRED", "BR_IND_TAKEN_RETIRED",
"BR_IMMED_SKIP_RETIRED", "BR_IMMED_TAKEN_RETIRED", "BR_SKIP_RETIRED"]}]}

Type

string

Default value**configure_v8_8_pmu_events_with_json**

"[DEPRECATED: Set configure_pmu_events_with_json to the same value instead] Configure v8.8 PMU events. Note : This param has high priority and overrides the setting of "has_v8_8_pmu_events" if both the params are provided. JSON schema for the parameter value is e.g. 1. {"all":false} 2. {"pmu_events":["BR_HINT_COND_RETIRED", "BR_COND_TAKEN_RETIRED", "BR_UNCOND_RETIRED", "BR_COND_RETIRED", "BRNL_TAKEN_RETIRED", "BRNL_IND_TAKEN_RETIRED", "BRNL_INDNR_TAKEN_RETIRED", "BRNL_IMMED_TAKEN_RETIRED", "BL_TAKEN_RETIRED", "BL_IND_TAKEN_RETIRED", "BL_IMMED_TAKEN_RETIRED"]}]}

Type

string

Default value**configure_v8_9_pmu_events_with_json**

"[DEPRECATED: Set configure_pmu_events_with_json to the same value instead] Configure v8.9 PMU events. Note : This param has high priority and overrides the setting of "has_v8_9_pmu_events" if both the params are provided. JSON schema for the parameter value is e.g. 1. {"all":false} 2. {"pmu_events":["ASE_SVE_RETIRED", "ASE_RETIRED", "VFP_RETIRED", "SVE_RETIRED", "CRYPTO_RETIRED", "SIMD_INST_RETIRED", "ASE_INST_RETIRED", "SVE_INST_RETIRED", "ASE_SVE_INST_RETIRED", "LD_ANY_RETIRED", "ST_ANY_RETIRED", "LDST_ANY_RETIRED", "DP_RETIRED"]}]}

Type

string

Default value**core_cache_protection**

core_cache_protection can change ERROFR, ERROPFGF and ERROPFGCTL fields. Possible values are: -1:Not implemented (by default), 0:Disabled, 1:Enabled

Type

int

Default value

0xffffffffffffffff

cpacr_trcds_behaviour

Behaviour of CPACR.TRCDIS/NSACR.NSTRCDIS when there is no CP14 ETM interface. 0, RAZ/WI. 2, implemented.

Type

int

Default value

0x2

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CP15SDISABLE2

Initialize to disable access to some CP15 registers (FEAT_CP15SDISABLE2).

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.DCZID-log2-block-size

Log2 of the block size cleared by DC ZVA instruction (as read from DCZID_EL0).

Type

int

Default value

0x8

cpuX.DCZVA_single_write

Execute the DCZVA as a single write

Type

bool

Default value

0x0

cpuX.MPIDR-override

Override of MPIDR value. If nonzero will override the MT, cluster and CPU ID bits in MPIDR.

Type

int

Default value

0x0

cpuX.RVBAR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.RVBAR32

Reset vector address in AARCH32 when VINITHI is not set and ignore_rvbar_in_aarch32 is set

Type

int

Default value

0x0

cpuX.SMPnAMP

Enable broadcast messages necessary for correct SMP operation at reset.

Type

bool

Default value

0x1

cpuX.TEINIT

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.aarch32_reset_from_impdef_addr

If PE resets into AArch32, Whether execution starts from IMPDEF address or hi/low vector

Type

bool

Default value

0x1

cpuX.ase-present

Set whether the model has been built with NEON support

Type

bool

Default value

0x1

cpuX.clock_divider

Clock divider ratio for asymmetric MP clocking.

Type

int

Default value

0x1

cpuX.clock_multiplier

Clock divider ratio for asymmetric MP clocking.

Type

int

Default value

0x1

cpuX.crypto_aes

AES instructions supported (requires CryptoPlugin to be loaded). 0, not implemented. 2, AES and PMULL instructions implemented (FEAT_AES, FEAT_PMULL).

Type

int

Default value

0x2

cpuX.crypto_sha1

SHA-1 instructions supported (requires CryptoPlugin to be loaded). 0, not implemented. 1, SHA1 instructions implemented (FEAT_SHA1).

Type

int

Default value

0x1

cpuX.crypto_sha256

SHA-256 instructions supported (requires CryptoPlugin to be loaded). 0, not implemented. 1, SHA256 instructions implemented (FEAT_SHA256).

Type

int

Default value

0x1

cpuX.cti-intack_mask

Set bits represent that the corresponding trigger requires software acknowledge via CTIINTACK

Type

int

Default value

0x1

cpuX.cti-number_of_claim_bits

Number of implemented bits in CTICLAIMSET

Type

int

Default value

0x0

cpuX.cti-number_of_triggers

Number of cti event triggers (default: 8, valid values: {3, 8-32})

Type

int

Default value

0x8

cpuX.dcache-size

L1 D-Cache size in bytes

Type

int

Default value

0x8000

cpuX.enable_crc32

CRC32 instructions supported. 0, not implemented. 1, CRC32 instructions implemented (FEAT_CRC32).

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.etm-present

Set whether the model has ETM support

Type

bool

Default value

0x1

cpuX.flash.enable

Enable flash by default after reset

Type

bool

Default value

0x0

cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x0

cpuX.force-fpsid-value

Value to override the FPSID value to

Type

int

Default value

0x0

cpuX.has_hcptr_tase

If false, HCPTR.TASE is RES0

Type

bool

Default value

0x1

cpuX.icache-size

L1 I-Cache size in bytes

Type

int

Default value

0x8000

cpuX.llpp.base

Sets the base address of Low Latency Peripheral Port

Type

int

Default value

0x0

cpuX.llpp.size

Sets the size of LLPP(in bytes)

Type

int

Default value

0x1000

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.number-of-breakpoints

Number of breakpoints.

Type

int

Default value

0x10

cpuX.number-of-context-breakpoints

Number of breakpoints that are context aware.

Type

int

Default value

0x10

cpuX.number-of-watchpoints

Number of watchpoints.

Type

int

Default value

0x10

cpuX.operation_bandwidth

Operation width for ARMv8.4 PMU extension

Type

int

Default value

0x1

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.semihosting-stderr_istty

Result for semihost istry call when argument is stderr

Type

bool

Default value

0x1

cpuX.semihosting-stdin_istty

Result for semihost istry call when argument is stdin

Type

bool

Default value

0x1

cpuX.semihosting-stdout_istty

Result for semihost istry call when argument is stdout

Type

bool

Default value

0x1

cpuX.semihosting-use_stderr

Send stderr from the simulated process to host stderr

Type

bool

Default value

0x0

cpuX.tcm-present

Disables the TCMs

Type

bool

Default value

0x0

cpuX.tcm-supports-exclusive

Whether TCM supports exclusive access

Type

bool

Default value

0x0

cpuX.tcm.a.base

Sets the base address of the ATCM. TCMs are implementation defined in the v8R architecture. TCMs are configurable on the model via parameter only

Type

int

Default value

0x0

cpuX.tcm.a.enable

Enable ATCM by default after reset. TCMs are implementation defined in the v8R architecture. TCMs are configurable on the model via parameter only

Type

bool

Default value

0x0

cpuX.tcm.a.size

Sets the size of the ATCM(in bytes). The size should be aligned with 4KB. The lower 12 bits will be masked out.

Type

int

Default value

0x4000

cpuX.tcm.a.stretch_clk

Whether ATCM clock stretched to occupy full cycle

Type

bool

Default value

0x0

cpuX.tcm.a.wait

ATCM accesses wait states

Type

int

Default value

0x0

cpuX.tcm.b.base

Sets the base address of the BTCM. TCMs are implementation defined in the v8R architecture. TCMs are configurable on the model via parameter only

Type

int

Default value

0x0

cpuX.tcm.b.enable

Enable BTCM by default after reset. TCMs are implementation defined in the v8R architecture. TCMs are configurable on the model via parameter only

Type

bool

Default value

0x0

cpuX.tcm.b.size

Sets the size of the BTCM(in bytes). The size should be aligned with 4KB. The lower 12 bits will be masked out.

Type

int

Default value

0x4000

cpuX.tcm.b.stretch_clk

Whether BTCM clock stretched to occupy full cycle

Type

bool

Default value

0x0

cpuX.tcm.b.wait

BTCM accesses wait states

Type

int

Default value

0x0

cpuX.tcm.c.base

Sets the base address of the CTCM. TCMs are implementation defined in the v8R architecture. TCMs are configurable on the model via parameter only

Type

int

Default value

0x0

cpuX.tcm.c.enable

Enable CTCM by default after reset. TCMs are implementation defined in the v8R architecture. TCMs are configurable on the model via parameter only

Type

bool

Default value

0x0

cpuX.tcm.c.size

Sets the size of the CTCM(in bytes). The size should be aligned with 4KB. The lower 12 bits will be masked out.

Type

int

Default value

0x2000

cpuX.tcm.c.stretch_clk

Whether CTCM clock stretched to occupy full cycle

Type

bool

Default value

0x0

cpuX.tcm.c.wait

CTCM accesses wait states

Type

int

Default value

0x0

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.unpredictable_WPMASKANDBAS

Constrained unpredictable handling of watchpoints when mask and BAS fields specified. 0, IGNOREMASK. 1, IGNOREBAS (default). 2, REPEATBAS8. 3, REPEATBAS.

Type

int

Default value

0x1

cpuX.vfp-dp-present

Whether double-precision floating point feature is implemented (FEAT_F64MM).

Type

bool

Default value

0x1

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

cpuX.vfp-traps

Implement support for trapping floating-point exceptions

Type

bool

Default value

0x1

cpuX.vfp-traps-show-all

Report all trapped floating-point exceptions in the syndrome when a combination occurs.

Type

bool

Default value

0x0

cpuX.wfet_early_or_delayed_timeout

WFET early or delayed timeout beyond the threshold value of CNTVCT_ELO in percentage.

Type

int

Default value

0x0

cpuX.wfit_early_or_delayed_timeout

WFIT early or delayed timeout beyond the threshold value of CNTVCT_ELO in percentage.

Type

int

Default value

0x0

dbg-bcr-reserved-behavior

This is the behavior of the reserved values of the BT field in DBGBCR. Possible values are: - 0 = Disabled. - 1 = BT[2] is ignored.

Type

int

Default value

0x1

dbg_rom_dap_addr

Debug ROM dap base address.

Type

int

Default value

0x0

dbgitr_buffer_size

Number of instructions which can be buffered before EDSCR.ITE is cleared

Type

int

Default value

0x0

dbgxvr_ress_is_stateful

Whether DBGWVR/DBGBVR.RESS returns last written value. if set to false, RESS returns sign extended value

Type

bool

Default value

0x0

dc_fault_unaligned_s1_device_s2_fwb

Whether takes an Alignment Fault caused by the memory type on a DC {ZVA,GZVA,GVA} if the stage 1 memory type is any Device memory type

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_bus_width_in_bytes

L1 D-Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-ways

L1 D-Cache number of ways (sets are implicit from size).

Type

int

Default value

0x2

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_bus_width_in_bytes

L1 D-Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcimva_requires_s2_write_permissions

Data-cache invalidate by MVA operations require stage 2 write permission (virtualised AArch32 guest).

Type

bool

Default value

0x0

debug_auth_signals_sampled_at_reset

Debug authentication signals can be configured as either sampled at reset only or at any time for External Root Debug.

Type

bool

Default value

0x0

debug_components_dap_address

Debug components ROM,ED,CTI,PMU,TRACE and TRBU base address for each core on debug bus. The "rom" field in the "cores" array are only allowed when 'debug_rom_is_flat' is false. JSON schema for the parameter value is:
 {"format":"all_addrs_are_absolute_wrt_debugbus", "cores": [{"rom":0x0, "ed":0x0, "cti":0x0, "pmu":0x0, "etm":0x0, "trbu":0x0}, {"rom":0x0, "ed":0x0, "cti":0x0, "pmu":0x0, "etm":0x0, "trbu":0x0}, {"rom":0x0, "ed":0x0, "cti":0x0, "pmu":0x0, "etm":0x0, "trbu":0x0}, {"rom":0x0, "ed":0x0, "cti":0x0, "pmu":0x0, "etm":0x0, "trbu":0x0}]}

Type

string

Default value**debug_components_mmap_address**

Debug components ROM,ED,CTI,PMU,TRACE and TRBU base address for each core on system bus. The "rom" field in the "cores" array are only allowed when 'debug_rom_is_flat' is false. JSON schema for the parameter value is:
 {"format":"all_addrs_are_absolute_wrt_systembus", "cores": [{"rom":0x0, "ed":0x0, "cti":0x0, "pmu":0x0, "etm":0x0, "trbu":0x0}, {"rom":0x0, "ed":0x0, "cti":0x0, "pmu":0x0, "etm":0x0, "trbu":0x0}, {"rom":0x0, "ed":0x0, "cti":0x0, "pmu":0x0, "etm":0x0, "trbu":0x0}, {"rom":0x0, "ed":0x0, "cti":0x0, "pmu":0x0, "etm":0x0, "trbu":0x0}]}

Type

string

Default value**debug_entry_is_context_sync**

If true, Entry in debug state is Context sync. Exiting debug state is a context synchronising operation, but entering is not. However some cpu implementation can consider also the Entry in Debug state as a CSE

Type

bool

Default value

0x0

debug_rom_is_class_9

If true, present a debug ROM table as a class 9 device. Otherwise, use a class 1 ROM table.

Type

bool

Default value

0x0

debug_rom_is_flat

If true, present a debug ROM table recommended by ARMv8 Debug Architecture. Otherwise, use nested ROM tables.

Type

bool

Default value

0x0

def_mem_map

Default memory map in a json format which is: { "start-end_inclusive" : "attributes", "*" : "attributes" } Where the * represents the entire physical address range and must be provided and the attributes can be a combination of following. MemoryType - NORMAL, GRE, nGRE, nGnRE, nGnRnE Shareability - ISH, OSH, NSH InnerAttributes - IWB, IWT, INC OuterAttribute - OWB, OWT, ONC ExecuteNever - XN

Type

string

Default value

{ "*" : "NORMAL INC ONC OSH" }

def_mem_map_file_path

Path of file describing default memory map in json format. When a valid path is provided, the below parameter 'def_mem_map' will be ignored.

Type

string

Default value**default_inner_shareable**

shareability for default memory map regions which are shareable

Type

bool

Default value

0x0

delay_serror

Add a propagation delay of serror signal into the core

Type

int

Default value

0x0

dic-spi_count

Number of shared peripheral interrupts implemented

Type

int

Default value

0x40

disable_impdef_abort_on_ic_maintenance

Disable IMP_INTLATENCY_EL2.MMDVM/LLRAMDVM controlling the abort on IC maintenance on MM port/LLRAM

Type

bool

Default value

0x0

disable_sve_plugin

If true, SVE will not be implemented in this processor even if the plugin is loaded (FEAT_SVE).

Type

bool

Default value

0x0

disable_unknown_update_event_on_reset

Disables SYSREG_UPDATE event notification on reset for the registers whose bitfields are all reserved or resets to architecturally unknown value

Type

bool

Default value

0x0

dsb_accumulate_threshold

Limit the maximum number of observable DSB side effects which can be queued (e.g. TLBI), after which DSB sync will be done automatically.

Type

int

Default value

0x100

edpfr_ras_unknown_bits_read_as_0

If true then UNKNOWN bits in RAS field in EDPFR are read as 0

Type

bool

Default value

0x0

e10_can_access_imp_def_functionality

If not made UNDEF by imp_def_functionality_behaviour, ELO can access IMPLEMENTATION DEFINED registers and system instructions.

Type

bool

Default value

0x0

e13_trap_priority_when_secure_debug_disabled

Undef when secure debug is disabled (EDSCR.SDD == 1) && boolean IMPLEMENTATION_DEFINED 'EL3 trap priority when SDD == 1'

Type

bool

Default value

0x0

enable_address_contig_check

Check the input address range for the table entries that have the contiguous hint bit set.

Type

bool

Default value

0x0

enable_debug_auth_signals_config

Debug Authentication Signals DBGEN, SPIDEN (and if RME is enabled RLPIDEN and RTPIDEN) are configurable (default) or not configurable, (hardwired to 1). This parameter is the integer representation of a bitmap to enable configuration of these signals, with: - BIT[0] = DBGEN - BIT[1] = SPIDEN - BIT[2] = RLPIDEN - BIT[3] = RTPIDEN

Type

int

Default value

0xf

enable_lock_step

Whether the core is configured in Dual Core Lock Step mode (FEAT_DCLS).

Type

bool

Default value

0x0

enable_tlb_contig_check

Perform extra pagetable walks to check translation table entries that have the contiguous hint bit set.

Type

bool

Default value

0x0

enhanced_pac2_level

Implements Enhanced PAC2. 0: No EnhancedPAC2, 1: EnhancedPAC2 Only, 2: EnhancedPAC2 with FPAC, 3: EnhancedPAC2 with FPACCombined.

Type

int

Default value

0x0

error_record_feature_register

RAS feature register values. An array of JSON objects. The JSON schema for the array is: [{"ED":0x0, "IMPDEF_3_2":0x0, "UI":0x0, "FI":0x0, "UE":0x0, "CFI":0x0, "CEC":0x0, "RP":0x0, "DUI":0x0, "CEO":0x0, "CI":0x0, "TS":0x0, "INJ":0x0, "FRX":0x0, "UC":0x0, "UEU":0x0, "UER":0x0, "UEO":0x0, "DE":0x0, "CE":0x0, "Visibility":"Core"},other_feature_register_values]. Where ED,UI,FI,CE and UE have valid values between 0x0 - 0x3. CFI and DUI have valid values 0x0, 0x2 and 0x3. CEC has valid values 0x0,0x2 or 0x4. RP,CEO,INJ,FRX,UC,UEU,UER,UEO,DE has valid values 0x0 or 0x1. CI and TS has valid values of 0x0, 0x1 and 0x2. Visibility has valid values "Core" or "Cluster"

Type

string

Default value**error_record_feature_register_json_file**

File path to the RAS feature register values as JSON. The file uses the same format as the error_record_feature_register parameter value

Type

string

Default value

erxpfctl_res0_stateful_mask

Mask for stateful bits for ERXPFGCTL which are RES0

Type

int

Default value

0x0

exception_catch_before_software_step

Exception catch priority for the exception trapping form of exception catch (Armv8.2 or later, or exception_catch_type=0). If true, the exception catch debug event has higher priority than software step and halting step.

Type

bool

Default value

0x1

exception_catch_type

Type of exception catch (Armv8.0 - Armv8.1 only). 0, exception trapping. 1, non-exception trapping, higher priority than step. 2, non-exception trapping, lower priority than step.

Type

int

Default value

0x0

exclusive_monitor_clear_on_atomic_from_same_master

Exclusive monitors in the cluster will be cleared by a atomic by the same master to the monitored address.

Type

bool

Default value

0x1

exclusive_monitor_clear_on_store_from_same_master

Exclusive monitors in the cluster will be cleared by a store by the same master to the monitored address.

Type

bool

Default value

0x1

exclusive_monitor_clear_on_strex_address_mismatch

Exclusive monitors in the cluster will be cleared when a strex fails because the address does not match.

Type

bool

Default value

0x1

exclusive_monitor_clear_on_strex_success

Exclusive monitors in the cluster will be cleared when a strex succeeds.

Type

bool

Default value

0x1

exercise_stxr_fail

Reject a pseudo-random majority of exclusive store instructions

Type

bool

Default value

0x0

ext_abort_device_GRE_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_GRE_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_GRE_read_is_critical

Critical reporting of device-GRE read external aborts.

Type

bool

Default value

0x0

ext_abort_device_GRE_read_is_sync

Synchronous reporting of device-GRE read external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_device_read_is_sync.

Type

int

Default value

0x2

ext_abort_device_GRE_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_device_read_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_GRE_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_device_read_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_GRE_write_is_critical

Critical reporting of device-GRE write external aborts

Type

bool

Default value

0x0

ext_abort_device_GRE_write_is_sync

Synchronous reporting of device-GRE write external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_device_write_is_sync.

Type

int

Default value

0x2

ext_abort_device_GRE_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_device_write_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_GRE_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_device_write_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_read_is_critical

Critical reporting of device-nGRE read external aborts.

Type

bool

Default value

0x0

ext_abort_device_nGRE_read_is_sync

Synchronous reporting of device-nGRE read external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_device_read_is_sync.

Type

int

Default value

0x2

ext_abort_device_nGRE_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_device_read_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_device_read_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_write_is_critical

Critical reporting of device-nGRE write external aborts

Type

bool

Default value

0x0

ext_abort_device_nGRE_write_is_sync

Synchronous reporting of device-nGRE write external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_device_write_is_sync.

Type

int

Default value

0x2

ext_abort_device_nGRE_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_device_write_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_device_write_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_read_acquire_is_sync

Synchronous reporting of device read with acquire external aborts

Type

bool

Default value

0x0

ext_abort_device_read_is_critical

Critical reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x1

ext_abort_device_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_device_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_device_write_is_critical

Critical reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_device_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_device_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_fill_data

Returned data, if external aborts are asynchronous

Type

int

Default value

0xfdfdfdfcfcfdfdfd

ext_abort_normal_cacheable_read_is_critical

Critical reporting of normal write-back cacheable-read external aborts

Type

bool

Default value

0x0

ext_abort_normal_cacheable_read_is_sync

Synchronous reporting of normal write-back cacheable-read external aborts

Type

bool

Default value

0x1

ext_abort_normal_cacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_normal_cacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_normal_cacheable_write_is_critical

Critical reporting of normal write-back cacheable write external aborts

Type

bool

Default value

0x0

ext_abort_normal_cacheable_write_is_sync

Synchronous reporting of normal write-back cacheable write external aborts

Type

bool

Default value

0x0

ext_abort_normal_cacheable_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_normal_cacheable_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_normal_noncacheable_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_noncacheable_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_noncacheable_read_is_critical

Critical reporting of normal noncacheable-read external aborts

Type

bool

Default value

0x0

ext_abort_normal_noncacheable_read_is_sync

Synchronous reporting of normal noncacheable-read external aborts

Type

bool

Default value

0x1

ext_abort_normal_noncacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_normal_noncacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_normal_noncacheable_write_is_critical

Critical reporting of normal noncacheable write external aborts

Type

bool

Default value

0x0

ext_abort_normal_noncacheable_write_is_sync

Synchronous reporting of normal noncacheable write external aborts

Type

bool

Default value

0x0

ext_abort_normal_noncacheable_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_normal_noncacheable_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_normal_wt_cacheable_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_wt_cacheable_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_wt_cacheable_read_is_critical

Critical reporting of normal write-through cacheable-read external aborts

Type

bool

Default value

0x0

ext_abort_normal_wt_cacheable_read_is_sync

Synchronous reporting of normal write-through read external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_normal_cacheable_read_is_sync.

Type

int

Default value

0x2

ext_abort_normal_wt_cacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_normal_cacheable_read_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_wt_cacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_normal_cacheable_read_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_wt_cacheable_write_is_critical

Critical reporting of normal write-through write external aborts

Type

bool

Default value

0x0

ext_abort_normal_wt_cacheable_write_is_sync

Synchronous reporting of normal write-through write external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_normal_cacheable_write_is_sync.

Type

int

Default value

0x2

ext_abort_normal_wt_cacheable_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_normal_cacheable_write_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_wt_cacheable_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_normal_cacheable_write_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_prefetch_device_GRE_read_is_critical

Critical reporting of external aborts generated by device-GRE instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_device_GRE_read_is_sync

Behaviour of external aborts generated by device-GRE instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_prefetch_device_nGRE_read_is_critical

Critical reporting of external aborts generated by device-nGRE instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_device_nGRE_read_is_sync

Behaviour of external aborts generated by device-nGRE instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_prefetch_device_read_is_critical

Critical reporting of external aborts generated by device-nGnRE instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_device_read_is_sync

Behaviour of external aborts generated by device-nGnRE instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_prefetch_is_critical

Critical reporting of external aborts generated by normal writeback cacheable instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_is_sync

Behaviour of external aborts generated by normal writeback cacheable instruction fetches. 0, asynchronous abort. 1, synchronous abort.

Type

bool

Default value

0x1

ext_abort_prefetch_noncacheable_read_is_critical

Critical reporting of external aborts generated by normal noncacheable instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_noncacheable_read_is_sync

Behaviour of external aborts generated by normal noncacheable instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_prefetch_so_read_is_critical

Critical reporting of external aborts generated by device-nGnRnE instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_so_read_is_sync

Behaviour of external aborts generated by device=nGnRnE instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_prefetch_wt_cacheable_read_is_critical

Critical reporting of external aborts generated by normal writethrough cacheable instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_wt_cacheable_read_is_sync

Behaviour of external aborts generated by normal writethrough cacheable instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_so_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffffffff

ext_abort_so_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffffffff

ext_abort_so_read_is_critical

Critical reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x1

ext_abort_so_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_so_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_so_write_is_critical

Critical reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x1

ext_abort_so_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_ttw_cacheable_read_is_critical

Critical reporting of TTW cacheable read external aborts

Type

bool

Default value

0x0

ext_abort_ttw_cacheable_read_is_sync

Synchronous reporting of TTW cacheable read external aborts

Type

bool

Default value

0x1

ext_abort_ttw_cacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_ttw_cacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_ttw_noncacheable_read_is_critical

Critical reporting of TTW noncacheable read external aborts

Type

bool

Default value

0x0

ext_abort_ttw_noncacheable_read_is_sync

Synchronous reporting of TTW noncacheable read external aborts

Type

bool

Default value

0x1

ext_abort_ttw_noncacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_ttw_noncacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_ttw_wt_cacheable_read_is_critical

Critical reporting of TTW write-through cacheable read external aborts

Type

bool

Default value

0x0

ext_abort_ttw_wt_cacheable_read_is_sync

Synchronous reporting of TTW write-through cacheable read external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_ttw_cacheable_read_is_sync.

Type

int

Default value

0x2

ext_abort_ttw_wt_cacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_ttw_cacheable_read_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_ttw_wt_cacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_ttw_cacheable_read_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

external_oslar_access_disabled_by_authentication

If true, external accesses to OSLAR, when external debugging is not enabled, will generate an error (FEAT_Debugv8p2).

Type

bool

Default value

0x0

fault_on_nT_bit_set

Whether block translation table entries with the nT bit set should always fault. Only applies when changing_block_size_without_bbm_support_level is 1 or higher.

Type

bool

Default value

0x1

fault_unalign_to_unsupported_access

If has_unaligned_single_copy_atomicity is true, whether unaligned A64 atomic, exclusive and acquire/release instructions to non iWB-oWB or the access crossing a 16-byte boundary generate fault. Bits 0,1,2,3 should be set accordingly to enable the fault behaviour. bit 0: atomic access should fault, bit 1: exclusive access should fault, bit 2: acquire/release should fault, bit 3: the access crossing a 16-byte boundary should fault

Type

int

Default value

0x8

fault_unaligned_s1_device_s2_fwb

Whether unaligned fault with stage1 Device memory and final memory attribute forced to normal by FWB. 0, No fault. 1, will fault. 2 No fault if final Shareability is NSH.

Type

int

Default value

0x0

flash_protection_enable_at_reset

Enable flash memory protection after reset

Type

bool

Default value

0x0

force_align_pc

UNPREDICTABLE branch to non-word-aligned address in ARM state is forced to be aligned

Type

bool

Default value

0x0

fpcr_short_vector_raz

FPSCR and FPCR fields LEN and STRIDE are hardwired to 0

Type

bool

Default value

0x0

fpsr_res0_stateful_mask

Mask for stateful bits of FPSR which are RES0

Type

int

Default value

0x0

gic.GICC-offset

Offset from PERIPHBASE for GICC registers.

Type

int

Default value

0x2000

gic.GICD-offset

Offset from PERIPHBASE for GICD registers. Will be ignored when GICv3 CPU interface is enabled, as distributor is then external to the cluster.

Type

int

Default value

0x1000

gic.GICH-offset

Offset from PERIPHBASE for GICH registers.

Type

int

Default value

0x4000

gic.GICH-other-CPU-offset

Offset from PERIPHBASE for GICH registers for accessing other CPUs in the cluster. Set to 0 to disable.

Type

int

Default value

0x5000

gic.GICV-alias

Offset from PERIPHBASE for alias of GICV registers. When gicv2-only, if zero no alias will be created; if gicv2-only=0, the param is deprecated, when zero or unset an alias is created in the place mandated by the architecture (GICV-base+0xF000)

Type

int

Default value

0x0

gic.GICV-offset

Offset from PERIPHBASE for GICV registers.

Type

int

Default value

0x6000

gic.PERIPH-size

Size of registers based at PERIPHBASE that are considered to be owned by the GIC. Any accesses in the range PERIPHBASE to PERIPHBASE+gic.PERIPH-size-1 that do not match GIC registers will be treated as RAZ/WI.

Type

int

Default value

0x8000

gic_iri.ARE-fixed-to-one

GICv2 compatibility is not supported and GICD_CTLR.ARE_* is always one

Type

bool

Default value

0x1

gic_iri.DPG-ARE-only

Limit application of DPG bits to interrupt groups for which ARE=1

Type

bool

Default value

0x0

gic_iri.DPG-bits-implemented

Enable implementation of interrupt group participation bits or DPG bits in GICR_CTLR

Type

bool

Default value

0x0

gic_iri.GICD-alias

In GICv2 mode: the base address for a 4k page alias of the first 4k of the Distributor page, in GICv3 mode: the base address of a 64KB page containing message based SPI signalling register aliases(0:Disabled)

Type

int

Default value

0x0

gic_iri.GICD_ITARGETSR-RAZWI

If true, the GICD_ITARGETS registers are RAZ/WI

Type

bool

Default value

0x0

gic_iri.GICD_PIDR

The value for the GICD_PIDR registers, if non-zero. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

gic_iri.GICR_PIDR

The value for the GICR_PIDR registers, if non-zero. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

`gic_iri.GITS_PIDR`

The value for the GITS_PIDR registers, if non-zero. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

`gic_iri.ICFGR-rsvd-bit`

If ARE=0, the value of reserved bits i.e. bit 0,2,4..30 of ICFGRn for n>0

Type

bool

Default value

0x1

`gic_iri.IIDR`

GICD_IIDR and GICR_IIDR value

Type

int

Default value

0x0

`gic_iri.IRI-ID-bits`

Number of bits used to represent interrupts IDs in the Distributor and Redistributors, forced to 10 when none of LPIs, extended SPIs or extended PPIs is supported

Type

int

Default value

0x10

`gic_iri.ITS-count`

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x0

gic_iri.SPI-count

Number of SPIs that are implemented.

Type

int

Default value

0x20

gic_iri.SPI-message-based-support

Distributor supports message based signaling of SPI

Type

bool

Default value

0x1

gic_iri.STATUSR-implemented

Determines whether the GICR_STATUSR register is implemented.

Type

bool

Default value

0x0

gic_iri.enable_protocol_checking

Enable/disable protocol checking at cpu interface

Type

bool

Default value

0x0

gic_iri.enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

gic_iri.has-two-security-states

If true, has two security states

Type

bool

Default value

0x0

gic_iri.irouter-default-mask

Default Mask value for IROUTER[32..1019] register in the form 'a.b.c.d'

Type

string

Default value

0.0.0.7

gic_iri.irouter-default-reset

Default Reset Value of IROUTER[32..1019] register in the form 'a.b.c.d' or *

Type

string

Default value

0.0.0.0

gic_iri.monolithic

Indicate that the implementation is not distributed

Type

bool

Default value

0x1

gic_iri.non-ARE-core-count

Maximum number of non-ARE cores; normally used to pass the cluster-level NUM_CORES parameter to the top-level redistributor.

Type

int

Default value

0x4

gic_iri.periph-size

Size in bytes allocated to internal GIC Distributor

Type

int

Default value

0x0

gic_iri.priority-bits

Number of implemented priority bits

Type

int

Default value

0x5

`gic_iri.processor-numbers`

Specify processor numbers (as appears in GICR_TYPER) in the form 0.0.0.0=0,0.0.0.1=1 etc.)
If not specified, will number processors starting at 0.

Type

string

Default value**`gic_iri.redistributor-offset`**

Offset from reg-offset where the Redistributors are accessible

Type

int

Default value

0x0

`gic_iri.redistributor-size`

Per Redistributor register space in bytes

Type

int

Default value

0x0

`gic_iri.reg-offset`

Offset from PERIPHBASE allocated to internal GIC Distributor

Type

int

Default value

0x0

`gic_iri.supports-shareability`

Device supports shareability attributes on outgoing memory bus (i.e. is modelling an ACElite port rather than an AXI4 port).

Type

bool

Default value

0x0

`gic_iri.virtual-lpi-support`

GICv4 Virtual LPIs and Direct injection of Virtual LPIs supported

Type

bool

Default value

0x0

`gic_iri.wakeup-on-reset`

Go against specification and start redistributors in woken-up state at reset. This allows software that was written for previous versions of the GICv3 specification to work correctly. This should not be used for production code or when the distributor is used separately from the core fast model.

Type

bool

Default value

0x0

`gicv3.A3-affinity-supported`

Whether a non-zero value for affinity at level 3 is supported.

Type

bool

Default value

0x0

`gicv3.BPR-min`

The minimum value for the GICC_BPR register (non-secure version will be 1 + this value).

Type

int

Default value

0x2

`gicv3.EOI-check-CPUID`

Check CPU ID specified for accesses to EOI registers (rather than just ending highest priority active interrupt).

Type

bool

Default value

0x0

`gicv3.EOI-check-ID`

Check Interrupt ID specified for accesses to EOI registers (rather than just ending highest priority active interrupt).

Type

bool

Default value

0x0

gicv3.EOI-deactivate-any-interrupt

Allow an EOI to deactivate interrupts that aren't the highest priority active interrupt (EOI-ignore-out-of-order must be false otherwise this is ignored).

Type

bool

Default value

0x0

gicv3.EOI-ignore-out-of-order

Ignore EOI writes that cannot end the highest priority active interrupt.

Type

bool

Default value

0x1

gicv3.FIQEn-RAO

GICC_CTLR.FIQEn is read as one, write insensitive

Type

bool

Default value

0x0

gicv3.IIDR_base

The base value for calculating the GICC_IIDR register value.

Type

int

Default value

0x43b

gicv3.LR-count

The number of implemented list registers.

Type

int

Default value

0x10

gicv3.PMHE-RAO-WI

ICC_CTLR_EL*.PHME is read as one, write insensitive

Type

bool

Default value

0x0

gicv3.PMHE-RAZ-WI

ICC_CTLR_EL*.PHME is read as zero, write insensitive

Type

bool

Default value

0x0

gicv3.PMHE-release-set-packet

if PHME is enabled, whether a SET packet is released by CPU Intf in Upstream Ack window.

Type

bool

Default value

0x0

gicv3.SRE-EL2-enable-RAO

When ICC_SRE_EL2.SRE is RAO/WI, makes ICC_SRE_EL2.Enable RAO/WI

Type

bool

Default value

0x0

gicv3.SRE-EL3-enable-RAO

When ICC_SRE_EL3.SRE is RAO/WI, makes ICC_SRE_EL3.Enable RAO/WI

Type

bool

Default value

0x0

gicv3.SRE-EL3-set-once

Restrict SRE EL3 to be set only once

Type

bool

Default value

0x0

gicv3.SRE-enable-action-on-mmap

Allowed values are: 0-SRE one allows mmap access. 1-SRE one disables mmap access. 2-SRE one makes mmap access RAZ-WI.

Type

int

Default value

0x0

gicv3.STATUSR-implemented

If GICv3 CPU interface is being used, this determines whether the STATUS registers are implemented

Type

bool

Default value

0x1

gicv3.VBPR-min

The minimum value for the GICV_BPR register (non-secure version will be 1 + this value).

Type

int

Default value

0x2

gicv3.VFIQEn-RAO

ICH_VMCR_EL2.VFIQEn is read as one, write insensitive

Type

bool

Default value

0x0

gicv3.cputnf-mmap-access-level

Allowed values are: 0-mmap access is supported for GICC,GICH,GICV registers. 1-mmap access is supported only for GICV registers. 2-mmap access is not supported.

Type

int

Default value

0x0

gicv3.dir-trap-support

The cpu supports separate trapping of ICC_DIR_EL1 to EL2

Type

bool

Default value

0x1

gicv3.el3_trap_priority_when_secure_debug_disabled

Undef to access priorities group register when secure debug is disabled

Type

bool

Default value

0x0

gicv3.extended-interrupt-range-support

Device has support for extended SPI/PPI ID ranges

Type

bool

Default value

0x0

gicv3.gicv2-only

Limit the GIC implementation to GICv2 features only

Type

bool

Default value

0x0

gicv3.idle-is-ff

For GICC/GICV RPR, when idle, return FF when true, minimum supported priority otherwise

Type

bool

Default value

0x1

gicv3.ignore-DIR-write-when-E0Imode-not-set

Ignore UNPREDICTABLE access to GICC_DIR register.

Type

bool

Default value

0x1

gicv3.interrupt-bypass-support

Interrupt bypass support, set to false for devices not supporting interrupt bypass

Type

bool

Default value

0x1

gicv3.local-SEIs

Generate SEI to signal internal issues

Type

bool

Default value

0x0

gicv3.local-VSEIs

Generate VSEI to signal internal issues

Type

bool

Default value

0x0

gicv3.physical-ID-bits

Number of physical ID bits implemented.

Type

int

Default value

0x10

gicv3.priority-bits

Number of priority bits implemented.

Type

int

Default value

0x5

gicv3.send-PMHE-command-only-when-priority-changes

Send PMHE upstream command to distributor only when write to ICC_PMR_EL1 changes the priority

Type

bool

Default value

0x0

gicv3.sgi-range-selector-support

Device has support for the Range Selector feature for SGI

Type

bool

Default value

0x0

gicv3.suppress-virtual-enables-comms

In GICv3 only mode, prevents the GIC CPUIF from communicating UpstreamWrite/VirtualEnables to the IRI

Type

bool

Default value

0x1

gicv3.virtual-ID-bits

Number of virtual ID bits implemented.

Type

int

Default value

0x10

gicv3.virtual-lpi-support

When GICv3 is supported, indicates a cut down CPUIF interface with no support of VLPI (GICv3 only) when false

Type

bool

Default value

0x1

gicv3.virtual-priority-bits

Number of virtual priority bits implemented.

Type

int

Default value

0x5

gicv3.without-DS-support

GICv3 CPU interfaces do not support disabling security in the distributor (GICD_CTLR.DS=1)

Type

bool

Default value

0x0

gicv4.mask-virtual-interrupt

If true, virtual interrupts can be masked from being reported to virtual CPU interface by setting ICH_HCR_EL2.DVIM 1. No control otherwise.

Type

bool

Default value

0x0

global_debug_rom.ROMDEVID

Value of Debug Rom Device Identification Register

Type

int

Default value

0x0

global_debug_rom.ROMPIDR

Value of Debug Rom Peripheral Identification Register

Type

int

Default value

0x4000bb000

global_debug_rom.ROMPRIDR0

Value of Debug ROM Power RequestID Register

Type

int

Default value

0x1

hardware_translation_table_update_implemented

Implement hardware translation table updates from ARMv8R-64.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has-gicv4.1

GICv4.1 is enabled, and all the features with GICv4.1 are implemented (FEAT_GICv4p1).

Type

bool

Default value

0x0

has_16bit_asids

Enable 16-bit ASIDs.

Type

bool

Default value

0x1

has_16bit_vmids

Implement support for 16-bit VMIDs from ARMv8R-64.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_16k_granule

Implement the 16k LPAE translation granule.

Type

bool

Default value

0x0

has_4k_granule

Implement the 4k LPAE translation granule.

Type

bool

Default value

0x1

has_64k_granule

Implement the 64k LPAE translation granule.

Type

bool

Default value

0x1

has_aarch32_dbgdidr_etc

DBGDIDR, DBGDRAR, DBGDSAR exist even if EL1 doesn't implement AArch32

Type

bool

Default value

0x1

has_aarch64

All implemented exception levels can run in AArch64

Type

bool

Default value

0x0

has_bc

Implement Armv8.8 Hinted Conditional Branch (FEAT_HBC)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_ccidx

Implement the ARMv8R-64 CCSIDR Extension. Extending the ccsidr number of sets (FEAT_CCIDX).

Type

bool

Default value

0x0

has_cluster_l1cache_size

Whether core supports cluster level l1cache size

Type

bool

Default value

0x1

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x0

has_const_pac

Feature for singular selection of PAC field (FEAT_CONSTPACFIELD).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_cvadp_support

Implement instruction to support cache clean by deep persistence (DC CVADP) from ARMv8.5, can be selected for core implemented on any arch version starting ARMv8.2 (FEAT_DPB, FEAT_DPB2).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_debug_rom

If true, a debug ROM will be generated describing the cluster's debug components.

Type

bool

Default value

0x1

has_delayed_ctireg

Delay the functional effect of CTI register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_dbgreg

Delay the functional effect of external debug register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_mdscr_el1

Delay the functional effect of MDSCR_EL1 register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_oslar_el1

Delay the functional effect of OSLAR_EL1 register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_pmureg

Delay the functional effect of PMU register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_sysreg

Delay the functional effect of system register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_dgh

Implements Data Gathering Hint instruction from ARMv8.6 (FEAT_DGH).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_e0pd

Implement ARMv8-R64 feature to prevent unprivileged access to one half of the memory

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_edacr

Implement EDACR register

Type

bool

Default value

0x1

has_enhanced_pac

If pointer authentication is enabled then implement enhanced PAC.

Type

bool

Default value

0x0

has_exception_trapping_form_of_vector_catch

Implement the exception trapping form of vector catch debug event.

Type

bool

Default value

0x1

has_export_m_port

The interrupt distributor has an optional interrupt export port for routing interrupts to an external device

Type

bool

Default value

0x1

has_far_not_valid

Implements FnV bit in ESR_ELx and xFSR, FAR not valid for synchronous external aborts.

Type

bool

Default value

0x0

has_far_not_valid_dfsc

Implements FnV bit in ESR_ELx, FAR not valid for synchronous external aborts for Data Abort.

Type

bool

Default value

0x0

has_far_not_valid_ifsc

Implements FnV bit in ESR_ELx and xFSR, FAR not valid for synchronous external aborts for Instruction Abort.

Type

bool

Default value

0x0

has_flash

Flash Port present

Type

bool

Default value

0x0

has_flash_protection

Implement flash memory protection

Type

bool

Default value

0x0

has_fp16

Implement the half-precision floating-point data processing instructions from ARMv8R-64 (FEAT_FP16).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_generic_authentication

Implement ARMv8.3 generic authentication.

Possible values of this parameter are: - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_hardware_translation_table_update

Type of hardware translation table supported (when enabled by hardware_translation_table_update_implemented). 0, not implemented. 1, access bit updates implemented. 2, access bit updates and dirty bit mechanism implemented.

Type

int

Default value

0x2

has_internal_gic_iri

Is Internal GIC IRI implemented

Type

bool

Default value

0x0

has_itd

Implement the optional IT disable feature.

Type

bool

Default value

0x1

has_large_system_ext

Implement the ARMv8 Large System Extensions (FEAT_LSE).

Type

bool

Default value

0x0

has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8R-64 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_llpp

Low Latency Peripheral Port present

Type

bool

Default value

0x0

has_mpm

Implement max-power mitigation mechanism (MPMM)

Type

bool

Default value

0x0

has_no_os_double_lock

Do not implement the OS double-lock (FEAT_DoubleLock).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_par_bit10_razwi

Whether PAR_EL1[10] is RAZ/WI

Type

bool

Default value

0x0

has_partial_delayed_mdscr_el1

has_delayed_oslar_el1 only apply to some bits of MDSCR_EL1 (MDE, KDE, TDCC, SS).

Type

bool

Default value

0x0

has_pc_sample_based_profiling

If true, pc sample-based profiling is enabled (FEAT_PCSRv8, FEAT_PCSRv8p2).

Type

bool

Default value

0x1

has_per_cluster_debug_auth_ports

If true then the debug authentication ports i.e. spniden, niden, rpliden, rtpiden, dbgen, spiden are available per cluster

Type

bool

Default value

0x0

has_p12

Whether EL2 is implemented

Type

bool

Default value

0x1

has_pmc

Programmable MBIST controllers implemented

Type

bool

Default value

0x0

has_pmu

Implement the optional Performance Monitors Extension (FEAT_PMUv3). 0, Not Implemented. 1, Implemented. 2, PMU is IMPLEMENTATION_DEFINED, PMU version would be set to 0xF and would behave as if no PMU is implemented

Type

int

Default value

0x1

has_pointer_authentication

Implement ARMv8.3 pointer authentication (FEAT_PAuth).

Possible values of this parameter are: - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_prediction_invalidation_instructions

Implement execution and data prediction invalidation from ARMv8-R64 (FEAT_SPECRES).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_qarma3_pac

Supports QARMA3 pointer authentication algorithm (FEAT_PACQARMA3)

Type

bool

Default value

0x0

has_ras

Implements the ARMv8 RAS Extension. 0 = NO_RAS, 1 = MINIMAL_RAS, 2 = FULL_RAS (FEAT_RAS)

Type

int

Default value

0x0

has_ras_armv84_extension

Implement ARMv8R-64 RAS Extension (FEAT_RASv1p1).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_ras_double_fault

Implement ARMv8.4 RAS Double Fault Extension (FEAT_DoubleFault).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_restriction_on_speculative_data_loaded

Implements the ARMv8-R64 security feature (Restrictions on the effects of speculation) (FEAT_CSV3)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_self_hosted_trace_extension

Implement support for the Self-hosted Trace Extensions from ARMv8R-64 (FEAT_TRF).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_small_page_table

Implement small page table support which increases the maximum value of TxSZ field from ARMv8R-64 (FEAT_TTST). Note: will be unimplemented only if both has_small_page_table=0x0 and has_pl2=0x0.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_software_lock

Implement software lock in memory-mapped CTI, PMU, and external debug interfaces

Type

bool

Default value

0x1

has_speculation_barrier_inst

Implement speculation barrier instruction (SB) from ARMv8-R64 (FEAT_SB).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_speculative_sei

If true, the PE can generate SError interrupt exceptions from speculative reads of memory, including speculative instruction fetches

Type

bool

Default value

0x0

has_spp

Shared Peripheral Port present

Type

bool

Default value

0x0

has_stage2_ap_speculative_update

Speculative update of S2 AP bit on S1 TTW. 0 = No Update, 1 = Update, 2 = Update including for AT ops

Type

int

Default value

0x0

has_synchronous_load_atomics

Report asynchronous abort due to unsupported load atomics as synchronous (Cacheable)

Type

bool

Default value

0x1

has_synchronous_load_atomics_noncacheable

Report asynchronous abort due to unsupported load atomics as synchronous (Non-Cacheable)

Type

bool

Default value

0x1

has_synchronous_store_atomics

Report asynchronous abort due to unsupported store atomics as synchronous (Cacheable)

Type

bool

Default value

0x0

has_synchronous_store_atomics_noncacheable

Report asynchronous abort due to unsupported store atomics as synchronous (Non-Cacheable)

Type

bool

Default value

0x0

has_tlb_conflict_abort

Detected inconsistent TLB content generate aborts.

Type

bool

Default value

0x0

has_tlb_pa_caching

Whether intermediate caching of translation table walks might include NonCoherent caches of previous valid walks. 0, NonCoherent caches might be included. 1, No NonCoherent caches included (FEAT_nTLBPA)

Type

bool

Default value

0x0

has_unsupported_exclusive_fault

Report unsupported exclusive access with Unsupported Exclusive fault status (otherwise use external abort)

Type

bool

Default value

0x1

has_v8_4_pmu_extension

Implement PMU extension from ARMv8.4 (FEAT_PMUv3p4).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_5_debug_over_power_down

Implement ARMv8.5 Debug over powerdown

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8R-64 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_v8_6_pmu_events

Implements PMU events from ARMv8.6

Possible values of this parameter are: - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_writebuffer

Implement write accesses buffering before L1 cache. May affect ext_abort behaviour.

Type

bool

Default value

0x0

hcptr_tta_behaviour

Behaviour of HCPTR.TTA when there is no CP14 ETM interface. 0, RAZ/WI. 1, RAO/WI. 2, stateful.

Type

int

Default value

0x2

hcr_el2_miocnce_is_rw

If true, HCR_EL2.MIOCNCE is treated as R/W instead of RAZ/WI

Type

bool

Default value

0x0

hcr_swio_res1

Whether HCR.SWIO and/or HCR_EL2.SWIO are RES1.

Type

bool

Default value

0x0

hsr_uncond_cc

Condition codes reported in HSR as AL if it passes

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-log2linelen

If nonzero, Log2 of the instruction cache line length in bytes (valid values in range 4-8). Otherwise the value of cache-log2linelen is used.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-nprefetch

Number of next sequential instruction cache lines to prefetch. This is only used when icache-prefetch_enabled=true.

Type

int

Default value

0x1

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-prefetch_level

0 based cache level at which instructions are pre-fetched. This is only used when icache-prefetch_enabled=true.

Type

int

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_bus_width_in_bytes

L1 I-Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

icache-ways

L1 I-Cache number of ways (sets are implicit from size).

Type

int

Default value

0x2

ignore_tag_check_dcc_load_store_in_ma_mode_when_tco_is_disabled

Constrained unpredictable behavior for reads/writes to external debug interface DTR regs in memory access mode when PSTATE.TCO is 0. If true, tag check is ignored else, tag check is performed if required.

Type

bool

Default value

0x0

imp_def_functionality_behaviour

Behaviour of IMPLEMENTATION DEFINED registers and system instructions. 0, UNDEF. 1, RAZ/WI.

Type

int

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

internal_vgic

Instantiate VGIC peripheral in this processor

Type

bool

Default value

0x0

is_debug_state_pmu_snapshot_allowed

If true, PMU snapshot is allowed in debug state.

Type

bool

Default value

0x1

is_first_pcsr_sample_ignored

If true, First read of PMPCSR register after reset returns 0xFFFFFFFF

Type

bool

Default value

0x0

is_serror_edge_triggered

If true, SError is edge-triggered. Otherwise, its level-triggered.

Type

bool

Default value

0x1

is_uniprocessor

Value for the U bit in MPIDR. true disables L1 cache coherency protocols

Type

bool

Default value

0x0

isb_is_branch

If true, ISB is traced as an immediate branch. This allows to count ISB as a branch in debug extensions (e.g. PMU)

Type

bool

Default value

0x0

ish_is_osh

Whether Innershareable is same as OuterShareable

Type

bool

Default value

0x0

itd_conditional_instructions_are_32bit

When SCTLR_ELx.ITD=1, an IT instruction plus a T16 instruction are considered a single 32bit conditional instruction.

Type

bool

Default value

0x0

jidr_is_undef_at_el0

If true, JIDR register access is UNDEF at EL0

Type

bool

Default value

0x0

jmcr_is_undef_at_el0

If true, JMCR register access is UNDEF at EL0

Type

bool

Default value

0x0

joscr_is_undef_at_el0

If true, JOSCR register access is UNDEF at EL0

Type

bool

Default value

0x0

l1cache_has_r52_cache_policy

Whether l1cache has r52 cache policy

Type

bool

Default value

0x0

l1cache_has_rsvd_flash_ways

Whether l1 cache segregates ways for flash and AXI data.

Type

bool

Default value

0x0

l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-mpamf.arch_major_ver

L3 Cache MPAMF_AIDR architecture major version

Type

int

Default value

0x0

l3cache-mpamf.arch_minor_ver

L3 Cache MPAMF_AIDR architecture minor version

Type

int

Default value

0x0

l3cache-mpamf.has_esr

L3 Cache's MPAMF_ESR, MPAMF_ECR, and MPAM error handling implemented.

Type

bool

Default value

0x0

l3cache-mpamf.has_extd_esr

L3 Cache's MPAMF_ESR is 64-bits.

Type

bool

Default value

0x0

l3cache-mpamf.has_impl_idr

L3 Cache's MPAMF_IMPL_IDR is present.

Type

bool

Default value

0x0

l3cache-mpamf.has_mbwu_long_counter

L3 Cache has long MBWU counter and capture registers.

Type

bool

Default value

0x0

l3cache-mpamf.has_mpamfidr_ext

MPAMF_IDR.EXT support

Type

bool

Default value

0x0

l3cache-mpamf.has_partid_nrw

Narrowing part ID register is present. This is global rather than per-instance.

Type

bool

Default value

0x0

l3cache-mpamf.has_priority_partitioning

The selected resource has priority partitioning described in MPAMF_PRI_IDR.

Type

bool

Default value

0x0

l3cache-mpamf.has_prod_id

L3 Cache MPAMF_IIDR product ID supported.

Type

int

Default value

0x0

l3cache-mpamf.has_prod_rev

L3 Cache MPAMF_IIDR product REVISION supported.

Type

int

Default value

0x0

l3cache-mpamf.has_prod_var

L3 Cache MPAMF_IIDR product VARIANT supported.

Type

int

Default value

0x0

l3cache-mpamf.has_ris

L3 Cache has resource instance selection support.

Type

bool

Default value

0x0

l3cache-mpamf.max_partid_ns

L3 Cache Maximum value of non-secure PARTID supported.

Type

int

Default value

0xffff

l3cache-mpamf.max_partid_rl

L3 Cache Maximum value of realm PARTID supported for RME implementations.

Type

int

Default value

0xffff

l3cache-mpamf.max_partid_rt

L3 Cache Maximum value of root PARTID supported for RME implementations.

Type

int

Default value

0xffff

l3cache-mpamf.max_partid_s

L3 Cache Maximum value of secure PARTID supported.

Type

int

Default value

0xffff

l3cache-mpamf.max_pmg_ns

L3 Cache Maximum value of non-secure PMG supported.

Type

int

Default value

0xff

l3cache-mpamf.max_pmg_rl

L3 Cache Maximum value of realm PMG supported for RME implementations.

Type

int

Default value

0xff

l3cache-mpamf.max_pmg_rt

L3 Cache Maximum value of root PMG supported for RME implementations.

Type

int

Default value

0xff

l3cache-mpamf.max_pmg_s

L3 Cache Maximum value of secure PMG supported.

Type

int

Default value

0xff

l3cache-mpamf.mbwu_long_counter_width

L3 Cache long MBWU counter width. 0: 63 bits, 1: 44 bits.

Type

int

Default value

0x0

l3cache-mpamf.no_impl_msmon

L3 Cache's MPAMF_IMPL_IDR does not describe resource monitors.

Type

bool

Default value

0x0

l3cache-mpamf.no_impl_part

L3 Cache's MPAMF_IMPL_IDR does not describe resource partitioning controls.

Type

bool

Default value

0x0

l3cache-mpamf.ris_max

L3 Cache's largest resource instance selector value defined.

Type

int

Default value

0x0

l3cache-mpamf_base

L3 Cache memory mapped MPAM registers base address

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_bus_width_in_bytes

L3 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x0

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_bus_width_in_bytes

L3 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

legacy_combining_exc_catch_trace

Whether exception catch is traced as part of exception entry/exit in same cycle

Type

bool

Default value

0x1

ls64_ignore_s1_unpred_memattr_transformation

If true, stage 1 unpredictable memory attribute transformations are ignored for FEAT_LS64 single-copy atomic 64-byte load/store instructions' (FEAT_LS64, FEAT_LS64_V, FEAT_LS64_ACCDATA).

Type

bool

Default value

0x0

ls64_memtype_check_use_combined_memattr

If true, FEAT_LS64 single-copy atomic 64-byte load/store instructions' memory attributes check is done on the combined memory attributes at the end of all enabled translation stages.

Type

bool

Default value

0x0

mdrar_el1_res0

MDRAR_EL1 is RES0.

Type

bool

Default value

0x0

memory.ext_slave_base

Base address of Slave Port. Each core's region will offset by ext_slave_size_per_core

Type

int

Default value

0x0

memory.ext_slave_size_per_core

Size of Slave region for each core

Type

int

Default value

0x0

memory.flash_base

Base address of Flash

Type

int

Default value

0x0

memory.flash_size

Size of the Flash RAM

Type

int

Default value

0x0

memory.has_llram

Low-Latency RAM present

Type

bool

Default value

0x0

memory.l2_cache.is_inner_cacheable

L2 cache obeys inner cacheable attributes (rather than outer cacheable attributes)

Type

bool

Default value

0x1

memory.l2_cache.is_inner_shareable

L2 cache obeys inner shareable attributes (rather than outer shareable attributes)

Type

bool

Default value

0x1

memory.llram_base

Base address of LLRAM

Type

int

Default value

0x0

memory.llram_enable_at_reset

Whether llram is enabled at reset

Type

bool

Default value

0x0

memory.llram_shared

Controls the Low-Latency RAM's sharability attribute.

Type

bool

Default value

0x0

memory.llram_size

Size of the LLRAM

Type

int

Default value

0x0

memory.scu_present

L1 Caches are coherent

Type

bool

Default value

0x1

memory.transmit_vmid_in_user_flags

Transmit VMID in transaction attributes

Type

bool

Default value

0x0

mixed_endian

Implement support for mixed endianness. 0, not supported. 1, supported at all exception levels. 2, supported at ELO only.

Type

int

Default value

0x1

mpidr_layout

Layout of MPIDR. 0 AFF0 is CPUID, 1 AFF1 is CPUID

Type

int

Default value

0x0

non_secure_vgic_alias_when_ns_only

If ! has_el3 and only non-secure side exists, then the normal position of the VGIC is a secure alias. If this parameter is non-zero then in addition a non-secure alias of the VGIC will be placed at this position (aligned to 32 KB).

Type

int

Default value

0x0

num_protection_regions_s1

Number of v8-R protection regions

Type

int

Default value

0x20

num_protection_regions_s2

Number of v8-R hyp protection regions

Type

int

Default value

0x20

num_spi

Number of interrupts (SPI) into the internal GIC controller

Type

int

Default value

0x20

number_of_error_records

Cores Number of Error records supported for RAS

Type

int

Default value

0x0

page_based_hardware_attributes

Implement the page based hardware attributes from ARMv8R-64. This parameter indicates which page table bits are available for hardware, where bits[3:0] correspond to PTE[62:59] and to TCR_ELx.HWUnyy (FEAT_HPDS2).

Type

int

Default value

0x0

par_ns_set_unknown_bit

Whether NS bit of PAR is set/clear when executing AT to perform non-secure regime translation. When true, NS is set to 1 else 0.

Type

bool

Default value

0x1

per_core_master_supported

If master port from each core is exposed out of cluster

Type

bool

Default value

0x0

pfr1_csv2_frac

Fractional revision number ID_AA64PFR1_EL1.CSV2_frac when ID_AA64PFR0_EL1.CSV==1 for CSV2 extension (FEAT_CSV2_1p1, FEAT_CSV2_1p2)

Type

int

Default value

0x0

pmb_idr_external_abort

Describes how the PE manages External aborts on writes made by the Statistical Profiling Extension to the Profiling Buffer. 0, External abort is reported to SPE, From Armv8.8 and Armv9.3, the value 0 is not permitted. 1, External abort is ignored. 2, The External abort generates an SError and the error is not reported to SPE.

Type

int

Default value

0x0

pmb_idr_flag_updates

Defines whether the address translation performed by the Profiling Buffer manages the Access Flag and dirty state

Type

bool

Default value

0x1

pmcr_disable_events_export

If true, export for PMU events is disabled. This configures PMCFGR.EX field.

Type

bool

Default value

0x1

pmmir_ell_bus_slots

Largest value by which BUS_ACCESS can increment over BUS_CYCLES cycles.

Type

int

Default value

0x0

pmmir_ell_bus_width

Width, in bytes, of accesses counted by BUS_ACCESS

Type

int

Default value

0x0

pms_idr_max_size

Defines largest size for a single SPE record (rounded up to a power of 2)

Type

int

Default value

0x6

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x8

pmu_cycle_counter_counts_actual_cycles

If true and Timing annotation is enabled, PMU cycle counter counts actual cycles, otherwise counts instructions executed

Type

bool

Default value

0x0

pmu_has_chain_event

PMU (if present) implements event number 0x1e, CHAIN.

Type

bool

Default value

0x1

pseudo_fault_generation_feature_register

ARMv8.4 Standard Pseudo-fault generation feature register values. JSON schema for the parameter value is: [{"OF":false, "UC":false, "UEU":false, "UER":false, "UEO":false, "DE":false, "CE":0x0, "CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":false, "NA":false}, other_pseudo-fault_generating_features_register_values]. Where OF, UC, UEU, UER, UEO, DE, CI, ER, PN, AV, MV, SYN, and R have valid false(NOT_SUPPORTED) and true(FEATURE_CONTROLLABLE), where CE can have 0(NOT_SUPPORTED), 1(NONSPECIFIC_CE_SUPPORTED) and 3(TRANSIENT_OR_PERSISTENT_CE_SUPPORTED) and NA can have false(component fakes detection on next access) or true(component fakes detection spontaneously). Effective only when ERXFR's INJ field allows it or has_ras_fault_injection is true.

Type

string

Default value**pstate_ssbs_type**

Implement speculative store bypass safe feature from ARMv8.5. 0, Not supported. 1, Supported without MSR/MRS access to SSBS (FEAT_SSBS). 2, fully supported (FEAT_SSBS2).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

ram_protection_enable_at_reset

Enable TCM, L1Cache memory protection after reset

Type

bool

Default value

0x0

randomize_unknowns_at_reset

Will fill in unknown bits in registers at reset with random value using register_reset_data as seed, it overrides scramble_unknowns_at_reset

Type

bool

Default value

0x0

ras_extra_configurations

Miscellaneous configurations for error records. An array of JSON objects. Note for ERXCTLR_EL1 register it only allows to define the mask value for the IMPDEF fields, ie bits [63:32] and bit 1, but its reset value applies on all fields. Note for ERXMISCn masks - these are 64 bit masks covering the 64 bit registers ERXMISCn_EL1. E.g. [{"Index": 0, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXMISC1_mask": 0x0, "ERXMISC1_reset": 0x0, "ERXMISC2_mask": 0x0, "ERXMISC2_reset": 0x0, "ERXMISC3_mask": 0x0, "ERXMISC3_reset": 0x0, "ERXCTLR_EL1_mask": 0x0, "ERXCTLR_EL1_reset": 0x0}, {"Index": 1, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXSTATUS_IERR_mask": 0x300}]

Type

string

Default value**ras_pfg_clock_mhz**

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0x18

register_reset_data

Data used to fill register bits when they become UNKNOWN at reset.

Type

int

Default value

0x0

register_reset_data_hi

Data used to fill the upper-half of 128-bit registers when the bits become UNKNOWN at reset.

Type

int

Default value

0x0

report_iside_cmo_ifsr

fault info for an iside cache maintenance operation is reported in the IFSR

Type

bool

Default value

0x1

report_second_access_align_fault_non_atomic_pair_access

If true, the non-atomic load/store pair accesses report the 2nd register as faulting address for an alignment fault. This is IMP-DEF behavior as defined in FEAT_LRCPC3.

Type

bool

Default value

0x0

report_second_access_mmu_fault_non_atomic_pair_access

If true, the non-atomic load/store pair accesses report the 2nd register as faulting address for an MMU fault. This is IMP-DEF behavior as defined in FEAT_LRCPC3.

Type

bool

Default value

0x0

reported_fp_revision

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored. Updates the FPSID.revision value

Type

int

Default value

0xffffffffffffffffffff

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reserved_HMC_SSC_PAC_treated_disabled

When DBG[B|W]CR.{HMC,SSC,PAC} bits configuration is reserved, this parameter controls whether breakpoints/watchpoints are treated as Disabled or not

Type

bool

Default value

0x0

restore_fpsr_on_trapped_fp_exception

If true, FPSR is restored to the value of the FPSR immediately before the instruction that generated the trapped floating-point exception.

Type

bool

Default value

0x0

restriction_on_speculative_execution

Implements the ARMv8-R64 security feature (Restrictions on the effects of speculation): 0: No disclosure whether branch targets trained in one context can affect speculative execution in a different context, 1: Branch targets trained in one context cannot affect speculative execution in a different hardware described context (SCXTNUM_ELx not supported), 2: Branch targets trained in one context cannot affect speculative execution in a different hardware described context (SCXTNUM_ELx supported) (FEAT_CSV2, FEAT_CSV2_2)

Type

int

Default value

0x0

rmr_always_implemented

Always implement RMR_ELx, RMR, or HRMR at the highest implemented exception level, even if that exception level cannot use both AArch32 and AArch64.

Type

bool

Default value

0x0

s1_align_memtype_fault_prio_more_than_s2_perm_fault_on_s1_walk

If true, s1 alignment fault has priority over s2 permission faults

Type

bool

Default value

0x1

scheduler_mode

Control the interleaving of instructions in this processor. 0, default long quantum. 1, low latency mode, short quantum and signal checking. 2, lock-breaking mode, long quantum with additional context switches near load-exclusive instructions. WARNING: This parameter is intended for validation purposes and may result in unwanted behaviour if altered!

Type

int

Default value

0x0

scr_nET_writeable

Whether SCR.nET is writeable. Writing to it is purely cosmetic (nET behavior not implemented)

Type

bool

Default value

0x0

scramble_unknowns_at_reset

Will fill in unknown bits in registers at reset with register_reset_data

Type

bool

Default value

0x1

serror_clear_delay

Delay for clearing of SError if SError is level-triggered, in cpu cycles.

Type

int

Default value

0x0

skip_trace_on_write_to_oseccr_el1_when_oslock_is_unlocked

If OSLSR_EL1.OSLK == 0, then OSECCR_EL1 returns an unknown value on reads and ignores writes. When true, also skips the traces on writes to OSECCR_EL1 when OSLSR_EL1.OSLK == 0.

Type

bool

Default value

0x0

spp.base

Sets the base address of Shared Peripheral Port

Type

int

Default value

0x0

spp.size

Sets the size of SPP(in bytes)

Type

int

Default value

0x1000

spsr_el3_is_mapped_to_spsr_mon

Whether SPSR_EL3 is mapped to AArch32 register SPSR_mon

Type

bool

Default value

0x0

spsr_m4_res0

Whether SPSR_ELx.M[4] bit should be RES0 for AARCH64 only implementations

Type

bool

Default value

0x0

stage12_tlb_size

If VMSA is supported at stage1, number of stage1+2 tlb entries. If instruction_tlb_size !=0, this is treated as dtlb size

Type

int

Default value

0x0

stage1_tlb_size

Number of stage1 only tlb entries.

Type

int

Default value

0x0

stage1_walkcache_size

Number of stage1 only walk cache entries.

Type

int

Default value

0x0

strex_fail_can_hit_watchpoint

If true, a strex fail can hit watchpoint

Type

bool

Default value

0x0

supports_multi_threading

Sets the MPIDR.MT bit. Setting this to true hints the the cluster is multi-threading compatible

Type

bool

Default value

0x0

swp_with_xzr_is_st_atomic

If true, swp with dest as xzr is treated as store atomic

Type

bool

Default value

0x1

take_ccfail_tsc_trap

When take_ccfail_undef=1 this parameter controls whether or not an SMC instruction that is trapped by HCR_EL2.TSC but fails its condition code check generates a trap to EL2.

Type

bool

Default value

0x0

take_ccfail_undef

UNDEF exception is taken even if condition code check fails

Type

bool

Default value

0x1

tidcp_traps_el0_undef_imp_def

TIDCP has priority over UNDEF for accesses to IMPLEMENTATION DEFINED functionality from EL0

Type

bool

Default value

0x1

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_or_ic_invalid_xt

Behavior of TLBI and IC instructions that don't take Xt as an argument when Xt != 0b11111.
0: TLBI and IC not UNDEF, 1: TLBI UNDEF, IC not UNDEF, 2: TLBI not UNDEF, IC UNDEF, 3: TLBI and IC UNDEF

Type

int

Default value

0x0

trace_has_sysreg_access

ETM trace registers support access via system registers

Type

bool

Default value

0x1

trace_icc_registers_as_icv_when_redirected

If true, update trace with ICV, instead of ICC when ICV registers are accessed depending on the core state.

Type

bool

Default value

0x0

trace_physical_registers_when_host_virtualisation_enabled

When host virtualisation is enabled, trace sysreg accesses to physical register accessed (0=disabled, 1=Trace only ELR/SPSR_EL1 as ELR/SPSR_EL2, 2=Trace all redirected registers as physical registers)

Type

int

Default value

0x0

trap_dc_cmo_to_pou_if_nop

Whether traps to DC CMO operations to PoU are ignored if the same is treated as NOP.

Type

bool

Default value

0x1

trap_ic_cmo_to_pou_if_nop

Whether traps to IC CMO operations to PoU are ignored if the same is treated as NOP.

Type

bool

Default value

0x1

trap_reserved_group3_id_regs

Whether setting HCR_EL2.TID3 traps reserved group3 id registers.

Type

bool

Default value

0x0

treat-dcache-cmos-to-poc-as-nop

Whether dcache maintenance operations to the point of coherency are required for instruction to data coherence. 0 - Clean/Invalidate ops required, 1 - Clean/Invalidate ops not required and cannot generate faults, 2 - Clean/Invalidate ops not required but can generate faults

Type

int

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

treat-dcache-invalidate-as-clean-invalidate

Treat data cache invalidate operations as clean and invalidate.

Type

bool

Default value

0x0

treat-icache-cmos-to-pou-as-nop

If has_coherent_icache is true, whether instruction cache invalidation operations to PoU which are treated as NOP can generate fault. 0 - cannot generate faults, 1 - can generate faults

Type

int

Default value

0x0

treat_forced_normal_as_device_for_excl_atomics

Whether exclusive/atomic access is supported in same manner as access to device if stage1 is Device memory and final memory attribute forced to normal by FWB

Type

bool

Default value

0x0

treat_pld_as_nop

If true, treat PLD as NOP.

Type

bool

Default value

0x0

treat_pli_as_nop

If true, treat PLI as NOP.

Type

bool

Default value

0x0

treat_wfi_wfe_as_nop

If true, never go into wait state for WFI or WFE instructions.

Type

bool

Default value

0x0

truncate_pc_on_illegal_exception_return_to_aarch32

On Illegal ERET to AArch32, truncate PC to 32-bits

Type

bool

Default value

0x1

unification-level

Level of Unification Inner Shareable for the cache hierarchy

Type

int

Default value

0x1

unification-uniprocessor-level

Level of Unification Uniprocessor for the cache hierarchy

Type

int

Default value

0x1

unpred_LSE128_overlap

Constrained unpredictable behaviours for 128-bit LSE overlap. 1 Constraint_UNDEF, 2 Constraint_NOP

Type

int

Default value

0x1

unpred_clear_ISV_for_exception_before_software_step

Whether ESR_ELx.ISV bit is cleared/set, when it is constrained unpredictable due to a different exception before a software step exception.

Type

bool

Default value

0x0

unpred_edscr_ns_set_unknown_bit

Unknown(x) bit in NS field in EDSCR can be configure to 0 or 1.

Type

bool

Default value

0x0

unpred_edscr_rw_unknown_bits_read_as_1

Unknown(x) bits in RW field in EDSCR are read as 1 instead of 0.

Type

bool

Default value

0x0

unpred_edscr_status_read_as_no_syndrome

Controls the choice of EDSCR.STATUS bit-values, when it is constrained unpredictable behaviour due to a different exception before a halting step debug event.

Type

bool

Default value

0x0

unpred_extdbg_unknown_bits

Data used to fill only in UNKNOWN bit-fields of external debug registers e.g., EDPFR and EDDFR.

Type

int

Default value

0x0

unpred_load_single_reg_overlap_with_wb

Constrained unpredictable behaviours for single load with writeback(might impact certain load pair instructions) 0 Constraint_WBSUPPRESS, 1 Constraint_UNDEF, 2 Constraint_NOP

Type

int

Default value

0x0

unpred_mrsmsr_currentlymapped_undef

UNPREDICTABLE register access (accessible from current mode using different instruction) modeled as NOP when false and UNDEF when true

Type

bool

Default value

0x0

unpred_mrsmsr_protfailed_undef

UNPREDICTABLE register access (not accessible from current PL and security state) modeled as NOP when false and UNDEF when true

Type

bool

Default value

0x0

unpred_par_attr_returns_mair

If true, PAR_EL1.ATTR represents the memory attributes as per the MAIR value instead of the ones in the descriptor.

Type

bool

Default value

0x0

unpred_sctlr_c_0_taggable_behaviour

Controls unpredictable effects when SCTLTR_ELx.C=0 for a stage 1 translation regime on whether memory is treated as taggable. Values: 0=Tagged, 1=Untagged but forced to Tagged when FWB=1 and stage 2 restores WB, 2 = Untagged

Type

int

Default value

0x2

unpred_stage2_mpu_and_bg_disabled

Constrained unpredictable when stage2 MPU and background disabled. 0, Stage-2 level 0 translation fault(Default). 1, Unknown memory attributes.

Type

int

Default value

0x0

unpred_store_exclusive_base_overlap

Constrained unpredictable behaviours for store exclusive when s==n. 0 Constraint_NONE, 1 Constraint_UNDEF, 2 Constraint_NOP

Type

int

Default value

0x0

unpred_store_pair_and_single_reg_overlap_with_wb

Constrained unpredictable behaviours for pair and single store with writeback(doesn't cover store exclusive) 0 Constraint_NONE, 1 Constraint_UNDEF, 2 Constraint_NOP

Type

int

Default value

0x0

unpred_tlbi_not_in_monitor_mode

Constrained unpredictable behaviors for AArch32 TLBI instructions executed in secure privileged mode other than Monitor mode. 0: Preferred behavior (default), 1: UNDEF, 2: NOP, 3: execute as if had been executed in Monitor mode

Type

int

Default value

0x0

unpred_tsize_aborts

Behaviour when TSize is out of range. 0, force into range. 1, translation fault, forces unpred_tsize_pamax_aborts to 1.

Type

bool

Default value

0x0

unpred_tsize_pamax_aborts

Behaviour when stage 2 TSize exceeds the physical address size (or 40bits, from AArch32). 0, force into range. 1, translation fault. Ignored if unpred_tsize_aborts is 1.

Type

bool

Default value

0x0

unpredictable_exclusive_abort_memtype

Cause MMU abort if exclusive access is not supported in certain memory type (0=exclusives allowed in all memory types, 1=exclusives abort in Device memory types, 2=exclusives abort in any type other than WB inner cacheable)

Type

int

Default value

0x0

unpredictable_hvc_behaviour

HVC unpredictable behaviour. 0, UNDEF. 1, NOP.

Type

int

Default value

0x0

unpredictable_smc_behaviour

SMC unpredictable behaviour. 0, UNDEF. 1, NOP.

Type

int

Default value

0x0

unpredictable_wfet_and_wfit_behaviour

WFET and WFIT unpredictable behaviour in debug state. 0, UNDEFINED. 1, NOP.

Type

int

Default value

0x1

unsupported_atomic_fault_type

Type of fault reported on unsupported atomic access. 0 = external abort if any reported by interconnect, 1 = precise unsupported atomic fault, 2 = precise external abort, 3 = imprecise external abort.

Type

int

Default value

0x0

unsupported_hw_update_fault_type

Type of abort reported when hw update to descriptor is done using unsupported memtype (0=No abort, 1=IMPDEF abort caused by memtype, 2=Sync external abort)

Type

int

Default value

0x0

use_architectural_names

Use names SP/LR/PC instead of R13/R14/R15

Type

bool

Default value

0x0

use_stage1_sh_as_input_to_stage2

IMPDEF case of whether to use stage1 shareability or OuterShareable as input to stage2 if stage1 is Device memtype

Type

bool

Default value

0x0

use_tlb_contig_hint

Translation table entries with the contiguous hint bit set generate large TLB entries.

Type

bool

Default value

0x0

user_defined_rom_table_debug_power_config

User defined ROM Table debug power domains for ED,CTI,PMU and TRACE, and DBGPCR configuration. The "version" field and "cores" array are mandatory. The "dbgpcr" array, if provided, must contain unique integers in the range [0, 32) describing which debug power domains have power control implemented. The "rom" and "dbgpcr" fields in objects in the "cores" array are only allowed when 'debug_rom_is_flat' is false. All power domain ID fields ("rom", "ed/pmu", "cti", "etm") must be in the range [0, 32). The "ed/pmu" field is mandatory. Example JSON for a hierarchical debug ROM layout: '{"version": 0, "dbgpcr": [0, 1], "cores": [{"dbgpcr": [1, 31], "rom": 0, "ed/pmu": 0, "cti": 31, "etm": 1}, {"ed/pmu": 0}]}'

Type

string

Default value**vpu_datapath_width**

VPU data path width

Type

int

Default value

0x80

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

warn_unpredictable_in_v7

If true, behaviour which is unpredictable in V7 yet is predictable in V8 will produce a warning

Type

bool

Default value

0x1

watchpoint-log2secondary_restriction

log2 size of secondary restriction of FAR/EDWAR possible values on watchpoint hit for load/store operations.

Type

int

Default value

0x0

wfe_wakeup_delay

Configure WFE wakeup delay in CPU cycles

Type

int

Default value

0x0

wfi_wakeup_delay

Configure WFI wakeup delay in CPU cycles

Type

int

Default value

0x0

wp_ignores_dbm_update

If true, dbm update is ignored on watchpoint hit

Type

bool

Default value

0x0

3.5.2 AEMvACT

ARM AEM A-Profile(MP) CPU component - number of cores configurable at runtime. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-177: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `d128_disabled_ps_resvd_size`
- `has_delayed_brbe_records`
- `hpfar_unknown_when_ipa_invalid`
- `mdrar_ell_res0`
- `mops_mismatched_page_crossing_access_unpred`
- `mpmm_accumulator_multiplier`
- `rme_full_is_tagged_nsh_treated_as_tagged`
- `tlbi_or_ic_invalid_xt`
- `tlbi_stall_enabled`
- `unpred_clear_ISV_for_exception_before_software_step`
- `unpred_mte_stzgm_tag_operation_before_data`
- `unpred_sctlr_c_0_taggable_behaviour`
- `use_stage1_sh_as_input_to_stage2`

Parameters removed:

- `atomic_fault_priority_s1`
- `atomic_fault_priority_s2`
- `atomic_fault_priority_s2_on_s1`
- `tlbi_invalid_xt_behave_as_undef`

Iris and MTI instances for AEMvACT

This model has the following Iris instances:

Table 3-178: AEMvACT Iris instances

InstanceName	ComponentName
AEMvACT	Cluster_ARM_AEM-A_MP
AEMvACT.AMU	PVBusLogger
AEMvACT.AMU.mapper	PVBusMapper
AEMvACT.DAP	PVBusLogger
AEMvACT.DAP.mapper	PVBusMapper
AEMvACT.DSU	DSU

InstanceName	ComponentName
AEMvACT.DSU.mpam_busslave	PVBusSlave
AEMvACT.MMAP	PVBusLogger
AEMvACT.MMAP.mapper	PVBusMapper
AEMvACT.acp_mapper	PVBusMapper
AEMvACT.cpu0	ARM_AEM-A_MP
AEMvACT.cpu0.UTLB	TLB
AEMvACT.cpu0.dtlb	TlbCadi
AEMvACT.cpu0.l1dcache	PVCache
AEMvACT.cpu0.l1dcache.upstream[0]	PVBusSlave
AEMvACT.cpu0.l1icache	PVCache
AEMvACT.cpu0.l1icache.upstream[0]	PVBusSlave
AEMvACT.ext_bus	PVBusLogger
AEMvACT.ext_bus.mapper	PVBusMapper
AEMvACT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder
AEMvACT.l2_cache	PVCache
AEMvACT.l2_cache.upstream[0]	PVBusSlave
AEMvACT.l2_cache.upstream[10]	PVBusSlave
AEMvACT.l2_cache.upstream[11]	PVBusSlave
AEMvACT.l2_cache.upstream[12]	PVBusSlave
AEMvACT.l2_cache.upstream[13]	PVBusSlave
AEMvACT.l2_cache.upstream[14]	PVBusSlave
AEMvACT.l2_cache.upstream[15]	PVBusSlave
AEMvACT.l2_cache.upstream[16]	PVBusSlave
AEMvACT.l2_cache.upstream[1]	PVBusSlave
AEMvACT.l2_cache.upstream[2]	PVBusSlave
AEMvACT.l2_cache.upstream[3]	PVBusSlave
AEMvACT.l2_cache.upstream[4]	PVBusSlave
AEMvACT.l2_cache.upstream[5]	PVBusSlave
AEMvACT.l2_cache.upstream[6]	PVBusSlave
AEMvACT.l2_cache.upstream[7]	PVBusSlave
AEMvACT.l2_cache.upstream[8]	PVBusSlave
AEMvACT.l2_cache.upstream[9]	PVBusSlave
AEMvACT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-179: AEMvACT MTI instances

InstanceName	ComponentName
AEMvACT	ARMv8Cluster
AEMvACT.AMU	PVBusLogger

InstanceName	ComponentName
AEMvACT.AMU.mapper	PVBusMapper
AEMvACT.DAP	PVBusLogger
AEMvACT.DAP.mapper	PVBusMapper
AEMvACT.DSU	DSU
AEMvACT.DSU.mpam_busslave	PVBusSlave
AEMvACT.MMAP	PVBusLogger
AEMvACT.MMAP.mapper	PVBusMapper
AEMvACT.acp_mapper	PVBusMapper
AEMvACT.cpu0	ARM_AEM-A_MP
AEMvACT.cpu0.UTLB	TLB
AEMvACT.cpu0.l1dcache	PVCache
AEMvACT.cpu0.l1dcache.upstream[0]	PVBusSlave
AEMvACT.cpu0.l1icache	PVCache
AEMvACT.cpu0.l1icache.upstream[0]	PVBusSlave
AEMvACT.ext_bus	PVBusLogger
AEMvACT.ext_bus.mapper	PVBusMapper
AEMvACT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder
AEMvACT.l2_cache	PVCache
AEMvACT.l2_cache.upstream[0]	PVBusSlave
AEMvACT.l2_cache.upstream[10]	PVBusSlave
AEMvACT.l2_cache.upstream[11]	PVBusSlave
AEMvACT.l2_cache.upstream[12]	PVBusSlave
AEMvACT.l2_cache.upstream[13]	PVBusSlave
AEMvACT.l2_cache.upstream[14]	PVBusSlave
AEMvACT.l2_cache.upstream[15]	PVBusSlave
AEMvACT.l2_cache.upstream[16]	PVBusSlave
AEMvACT.l2_cache.upstream[1]	PVBusSlave
AEMvACT.l2_cache.upstream[2]	PVBusSlave
AEMvACT.l2_cache.upstream[3]	PVBusSlave
AEMvACT.l2_cache.upstream[4]	PVBusSlave
AEMvACT.l2_cache.upstream[5]	PVBusSlave
AEMvACT.l2_cache.upstream[6]	PVBusSlave
AEMvACT.l2_cache.upstream[7]	PVBusSlave
AEMvACT.l2_cache.upstream[8]	PVBusSlave
AEMvACT.l2_cache.upstream[9]	PVBusSlave
AEMvACT.l2_flusher	AsyncCacheFlushUnit

AEMvACT contains the following CADI targets:

- ARM_AEM-A_MP

- Cluster_ARM_AEM-A_MP
- PVCache
- TlbCadi

About AEMvACT

AEMvACT implements all architectural features in Arm®v8-A and Armv9-A.

It provides parameters to enable or disable support for particular architectural features. Some of these parameters allow you to enable features in versions of the architecture earlier than the one in which they were introduced. They have the possible values:

- 0** Feature is not enabled.
- 1** Feature is enabled, but only if the model is set to implement a version of the architecture in which the feature is supported.
- 2** Feature is enabled, regardless of which architecture version the model implements.

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (crypto.dll or crypto.so) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Mapping architectural features to Fast Models releases

The following table shows which Fast Models releases support new architectural features, from Armv8.3-A onwards. Support is at EAC quality unless otherwise stated.

Table 3-180: Architectural features implemented in Fast Models

Architecture version	Feature	Fast Models version									
		11.26	11.25	11.24	11.23	11.22	11.21	11.20	11.19	11.18	11.17
Armv8.3-A	All. For details, see list after table.	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Armv8.4-A	All. For details, see list after table.	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Armv8.5-A	Memory Tagging Extension (MTE) functionality	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	MTE performance	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	All except MTE. For details, see list after table.	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Armv8.6-A	All. For details, see list after table.	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Armv8.7-A	Limited TLBI maintenance	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	All except TLBI maintenance. For details, see list after table.	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Architecture version	Feature	Fast Models version									
		11.26	11.25	11.24	11.23	11.22	11.21	11.20	11.19	11.18	11.17
Armv8.8-A	All	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Armv8.9-A	Guarded Call Stack, 128-bit page table descriptors, and PMU updates	✓	✓	✓	✓	✓	✓	✓	-	-	-
	All other features	✓	✓	✓	✓	✓	✓	✓	-	-	-
Armv9.0-A to Armv9.3-A	Branch Record Buffer Extension (BRBE).	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Trace Buffer Extension (TRBE) and Embedded Trace Extension (ETE). Reading TRBE registers was supported from 11.15.	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Realm Management Extension (RME) and Scalable Matrix Extension (SME).	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Scalable Matrix Extension 2 (SME2).	✓	✓	✓	✓	✓	✓	✓	✓	-	-
	All other features. For details, see the list below this table.	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Armv9.4-A	RME without Secure EL2.	✓	✓	✓	✓	✓	✓	-	-	-	-
	All other features.	✓	✓	✓	✓	✓	✓	✓	-	-	-
Armv9.5-A	All features.	✓	✓	✓	-	-	-	-	-	-	-

- Armv8.3-A features implemented include:
 - Pointer authentication
 - Nested virtualization
 - Advanced SIMD complex number support
 - Improved Javascript data type conversion support
 - Memory consistency model changes
 - Support for larger system-visible caches
 - Debug over powerdown
- Armv8.4-A features implemented include:

- Secure EL2
- Additional cryptographic hashes
- Activity monitors
- Enhanced support for nested virtualization
- Memory Partitioning and Monitoring (MPAM)
- Stage 2 forced write-back
- Dot product instructions
- Data independent timing instructions
- Large System Extensions (LSE)
- Support for TLB maintenance instructions and for TLB range instructions
- Translation Table Level (TTL)
- Enhancements to weaker release consistency
- Debug relaxations and extensions
- Flag manipulation instructions
- Armv8.5-A features implemented include:
 - Branch Target Indicators (BTI)
 - Enhanced Virtualization Traps (EVT)
 - Random Number Generator (RNG)
 - Flag Manipulation instructions
 - Floating-point round to integer
 - DC CVADP
- Armv8.6-A features implemented include:
 - General Matrix Multiply (GEMM)
 - EnhancedPAC2
 - FPAC
 - Data Gathering Hint
 - Additional PMU events
 - Enhanced Counter Virtualization
- Armv8.7-A features implemented include:
 - WFE and WFI with timeouts
 - Larger physical address for 4KB and 16KB translation granules
 - MTE Asymmetric Fault Handling
 - Enhancements to Privilege Access Never (PAN)
 - Enhancements to PMU and SPE
- Other Armv9-A features implemented include:

- Scalable Vector Extension version 2 (SVE2)
- Transactional Memory Extension (TME)

AEMvA core personalities

The AEM core personalities feature allows you to configure AEM instances in a platform to use the **IMPLEMENTATION_DEFINED** registers and **UNPREDICTABLE** behavior for a specific implementation of your choice at model startup.

Configuring an AEM with a core personality requires a license for both the AEM and the selected implementation.

Set the personality using either:

- The environment variable `FASTSIM_AEM_A_PROFILE`.
- The parameter `impdef_regs_and_unpred_from_implementation`.

The parameter allows you to configure different instances in the same platform with different personalities, including subclusters in a heterogeneous AEM. The environment variable takes precedence over the parameter and affects all instances of the AEM in the platform. Otherwise the two options function identically.

To see the available values for the environment variable or parameter, set either of them to the special value `list`. The model prints the list of available values and exits. An example value is `ARM_Cortex-X2`. Then set the parameter or environment variable to the required value.

Configuring a core personality only affects **IMPLEMENTATION_DEFINED** registers and **UNPREDICTABLE** behavior. In other respects, the cluster or subcluster still behaves like the AEM. For example, all parameters default to the AEM default values. Therefore, you must manually set parameters to valid values for the configured personality, as many of the defaults are incompatible with any given implementation.

To assist with this configuration, the AEM prints warnings for any parameter with an invalid value for the configured personality. The warning lists the parameter name and the valid value or range of values it can be set to for the selected personality.



Running the model with invalid parameter configurations for the selected personality can lead to unexpected behavior.

Ports for AEMvACT

Table 3-181: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcasttinner</code>	Signal	Slave	ACE defined pins.

Name	Protocol	Type	Description
broadcastouter	Signal	Slave	ACE defined pins.
cfgend[8]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clrexmonack	Signal	Master	Acknowledge handshake signal for the clrexmonreq signal
clrexmonreq	Signal	Slave	Signals the clearing of an external global exclusive monitor
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
CNTHPIRQ[8]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[8]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[8]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[8]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[8]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[8]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[8]	Signal	Master	Timer signals to SOC.
commirq[8]	Signal	Master	Interrupt signal from debug communications channel.
config64[8]	Signal	Slave	Register width after reset.
cp15sdisable[8]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[8]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
CRITICALIRQ[8]	Signal	Master	RAS Critical Error Interrupt.
cryptodisable[8]	Signal	Slave	Disable cryptography extensions after reset.
cti[8]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
cti0extin[4]	Signal	Slave	CTI trace inputs for core 0.
cti0extout[4]	Signal	Master	CTI trace outputs for core 0.
cti1extin[4]	Signal	Slave	CTI trace inputs for core 1.
cti1extout[4]	Signal	Master	CTI trace outputs for core 1.
cti2extin[4]	Signal	Slave	CTI trace inputs for core 2.
cti2extout[4]	Signal	Master	CTI trace outputs for core 2.
cti3extin[4]	Signal	Slave	CTI trace inputs for core 3.
cti3extout[4]	Signal	Master	CTI trace outputs for core 3.
cti4extin[4]	Signal	Slave	CTI trace inputs for core 4.
cti4extout[4]	Signal	Master	CTI trace outputs for core 4.
cti5extin[4]	Signal	Slave	CTI trace inputs for core 5.

Name	Protocol	Type	Description
cti5extout[4]	Signal	Master	CTI trace outputs for core 5.
cti6extin[4]	Signal	Slave	CTI trace inputs for core 6.
cti6extout[4]	Signal	Master	CTI trace outputs for core 6.
cti7extin[4]	Signal	Slave	CTI trace inputs for core 7.
cti7extout[4]	Signal	Master	CTI trace outputs for core 7.
ctidbgirq[8]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen[8]	Signal	Slave	External debug interface.
dbgnopwrdown[8]	Signal	Master	Debug no power down request.
dbgpwrdownack[8]	Signal	Master	Debug power down acknowledge.
dbgpwrdownreq[8]	Signal	Slave	Debug power down request.
dbgpwrupreq[8]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
ERRORIRQ[8]	Signal	Master	RAS Error Recovery Interrupt.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
external_trace_reset[8]	Signal	Slave	ETMv4 External Trace Reset signal.
FAULTIRQ[8]	Signal	Master	RAS Fault Handling Interrupt
fiq[8]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
fiq_nmi[8]	Signal	Slave	-
gicv3_redistributor_s[8]	GICv3Comms	Slave	GICv3 cpu interface ports.
hacdbsirq[8]	Signal	Master	Interrupt signal from the HACDBS unit.
irq[8]	Signal	Slave	This signal drives the CPUs interrupt handling.
irq_nmi[8]	Signal	Slave	-
irqs[224]	Signal	Slave	These signals drive the CPU's interrupt controller interrupt lines.
l2reset	Signal	Slave	This signal resets timer and interrupt controller.
memorymapped_amu_s	PVBus	Slave	External interface for amu.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[8]	Signal	Slave	External debug interface.
periphbase	Value_64	Slave	This port sets the base address of private peripheral region.
pmbirq[8]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[8]	Signal	Master	Interrupt signal from performance monitoring unit.

Name	Protocol	Type	Description
presetdbg[8]	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
rei[8]	Signal	Slave	Individual processor RAM Error Interrupt signal input.
reset[8]	Signal	Slave	Raising this signal will put the core into reset mode.
rlpiden[8]	Signal	Slave	External debug interface.
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
rtpiden[8]	Signal	Slave	External debug interface.
rvbar[8]	Value_64	Slave	Reset vector base address.
sei[8]	Signal	Slave	Per core System Error physical pins.
smpnamp[8]	Signal	Master	This signals AMP or SMP mode for each core.
spiden[8]	Signal	Slave	External debug interface.
spniden[8]	Signal	Slave	External debug interface.
standbywfe[8]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[8]	Signal	Master	This signal indicates if a core is in WFI state.
standbywfil2	Signal	Master	This signal indicated all cores and L2 are in a power down state
teinit[8]	Signal	Slave	This signal provides default exception handling state.
ticks[8]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trace_unit_reset[8]	Signal	Slave	ETMv4 Trace Unit Reset signal.
trbirq[8]	Signal	Master	Interrupt signal from the trace buffer unit.
vcpumntirq[8]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[8]	Signal	Slave	Virtual FIQ input. Note that the irq/fiq pins are wired directly to the core if there is no internal VGIC. If there is an internal VGIC then these are ignored.
vinithi[8]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[8]	Signal	Slave	Virtual IRQ input. Note that the irq/fiq pins are wired directly to the core if there is no internal VGIC. If there is an internal VGIC then these are ignored.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).

Name	Protocol	Type	Description
vsei[8]	Signal	Slave	Processor Virtual System Error Interrupt request.

Parameters for AEMvACT

ADFSR-AIFSR-implemented

ADFSR and AIFSR are implemented

Type

bool

Default value

0x0

AIDR

Value of AIDR_EL1 register.

Type

int

Default value

0x0

AMIIDR

Value of AMU Implementation Identification Register

Type

int

Default value

0x43b

AMPIDR

Value of AMU Peripheral Identification Register

Type

int

Default value

0x4000bb000

BPIMVA_causes_translation_lookup

Do a translation when BPIMVA instruction is executed (which may cause a translation fault).

Type

bool

Default value

0x0

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CCSIDR-L1D_override

If nonzero, override the value presented in CCSIDR for L1D (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L1I_override

If nonzero, override the value presented in CCSIDR for L1I (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L2_override

If nonzero, override the value presented in CCSIDR for L2 (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L3_override

If nonzero, allow L3 selection in CSSELR and present this value in CCSIDR (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

CTIPIDR

If non-zero, override the CTI Peripheral Identification Register

Type

int

Default value

0x0

CTR-L1Ip-override

If non-zero, override the L1Ip bits in CTR/CTR_ELO system register. This does not change the behaviour of the cache, only what is present in the CTR register.

Type

int

Default value

0x0

DBGBCR_BT_applies_RES0_before_valid_check

If true, RES0 behaviour is applied to DBGBCR(EL1).BT before checking for reserved values for this field.

Type

bool

Default value

0x1

DBGPIDR

If non-zero, override the Debug Peripheral Identification Register

Type

int

Default value

0x0

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

ERRIIDR

Value of RAS Implementation Identification Register

Type

int

Default value

0xd800143b

ERRPIDR

Value of RAS Peripheral Identification Register

Type

int

Default value

0x4100bbd80

ERXMISC0_mask

Write Mask for ERXMISC0 RAS Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

GMID-log2-block-size

Log2 of the block size accessed by STGM/LDGM/STZGM instructions

Type

int

Default value

0x4

ISV_set_to_0_for_stage2_synch_external_abort

Whether ESR_EL2.ISV is set to 0 on stage 2 synchronous external aborts

Type

bool

Default value

0x0

MIDR

Value of MIDR_EL1 register.

Type

int

Default value

0x410fd0f0

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

PA_SIZE

Physical address range supported For ARMv8.0 and ARMv8.1 this is limited to 48 bits (FEAT_LPA/FEAT_LPA3).

Type

int

Default value

0x28

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

PMCEID0

Performance Monitor Common Event ID Reg 0 value - 64 bit

Type

int

Default value

0xffffffff

PMCEID1

Performance Monitor Common Event ID Reg 1 value - 64 bit

Type

int

Default value

0xffffffff

PMSIDR.ArchInst

Defines whether architecture instruction sampling is implemented or not, if not only micro op sampling is implemented. Model only supports architecture instruction sampling, but allows ID register field to be configured.

Type

bool

Default value

0x1

PMSIDR.CRR

Defines whether call return branch records (FEAT_SPE_CRR) is implemented or not

Type

bool

Default value

0x0

PMSIDR.LDS

Defines whether data source for sampled load instruction is implemented or not. Model does not implement loaded data source, but allows ID register field to be configured.

Type

bool

Default value

0x0

PMUPIDR

If non-zero, override the PMU Peripheral Identification Register

Type

int

Default value

0x0

abort_execution_from_device_memory

Execution from device memory generates a prefetch abort.

Type

bool

Default value

0x0

advsimd_bf16_support_level

Implement BFloat16 operations from ARMv8.6. AArch64 Advanced SIMD and FP BFloat16 instructions are automatically enabled when has_arm_v8-6 is true. - 0, Not implemented. - 1, AArch64 Advanced SIMD and FP BFloat16 instructions only (FEAT_BF16). - 2, AArch32 Advanced SIMD and VFP BFloat16 instructions only (FEAT_AA32BF16). - 3, Both AArch64 Advanced SIMD and FP and AArch32 Advanced SIMD and VFP BFloat16 instructions.

Type

int

Default value

0x0

advsimd_i8mm_support_level

Implement Int8 matrix multiply operations from ARMv8.6. AArch64 Advanced SIMD and FP Int8 matrix multiply instructions are automatically enabled when has_arm_v8-6 is true. - 0, Not implemented. - 1, AArch64 Advanced SIMD and FP Int8 matrix multiply instructions only (FEAT_I8MM). - 2, AArch32 Advanced SIMD and VFP Int8 matrix multiply instructions only (FEAT_AA32I8MM). - 3, Both AArch64 Advanced SIMD and FP and AArch32 Advanced SIMD and VFP Int8 matrix multiply instructions (FEAT_I8MM, FEAT_AA32I8MM).

Type

int

Default value

0x0

advsimd_overread

AdvSIMD element load operations access all bytes of a 16-byte aligned window, even in Device memory

Type

bool

Default value

0x0

align_pc_on_branch_to_unaligned_pc_aarch32

Force PC align for branches to an unaligned PC counter in A32 state

Type

bool

Default value

0x0

align_pc_on_debug_exit_to_aarch32

Exit to AARCH32 state from debug state forces pc bit0 to 0

Type

bool

Default value

0x0

align_pc_on_illegal_exception_return_to_aarch32

Align PC when performing an illegal exception return from AArch64 to AArch32.

Type

bool

Default value

0x1

allow_s1_dbm_update_on_s2_mmu_fault

Whether s1 dirty bit update is done when s2 of ipa (not s1 ttw) generates mmu fault

Type

bool

Default value

0x1

amair_reg_rw_mask

RW mask for implementation-defined registers.

Type

int

Default value

0x0

amu_aux_type_fixed

Lists which AMU auxiliary registers that are fixed and to which event type. The JSON schema is: {fixed_aux_reg:evt_type, ...}. For example {"0":0x300} would make auxiliary register 0 fixed to event type 0x300

Type

string

Default value**amu_aux_voffset_mask**

If ARMv8.6 is implemented, each bit of the field, 0 to 15, when 1 indicates that the corresponding virtual offset register, AMEVCNTVOFF1<n>_EL2, is implemented

Type

int

Default value

0x0

amu_has_external_interface

Implement external memory-mapped access to system register of activity monitor unit from ARMv8.4.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

amu_has_sysreg_interface

Implement system register access to activity monitor unit from ARMv8.4.

Type

bool

Default value

0x1

amu_mmap_address

AMU base address for each core on system bus. 0 means the AMU is not mapped, otherwise the address must be 4KB aligned. JSON schema for the parameter value is: {"format": "all_addrs_are_absolute_wrt_systembus", "cores": [{"amu": 0x0}, {"amu": 0x0}, {"amu": 0x0}, {"amu": 0x0}]}

Type

string

Default value**amu_num_auxiliary_counters**

Number of AMU auxiliary counters implemented

Type

int

Default value

0x0

amu_reset_domain

Reset domain for activity monitor unit. 0, COLD_RESET. 1, WARM_RESET. 2, NONE.

Type

int

Default value

0x0

amu_version

Selects the activity monitor version implemented - 1, AMUv1 for Armv8.4 is implemented. - 2, AMUv1 for Armv8.6 is implemented (FEAT_AMUv1p1).

Type

int

Default value

0x1

apsr_read_restrict

At EL0, unknown bits of APSR are RAZ.

Type

bool

Default value

0x0

arm_v8_7_accelerator_support_level

Implements Armv8.7 accelerator support instructions LD/ST64B, ST64BV, ST64BV0, etc. (FEAT_LS64, FEAT_LS64_V, FEAT_LS64_ACCDATA)

Type

int

Default value

0x0

atomic_memtype_fault_priority

This parameter describes the priority of unsupported atomic/exclusive memtype fault w.r.t alignment and permission fault. 0, BEFORE_ALIGN_MEM_FAULT. 1, AFTER_ALIGN_BEFORE_PERM_FAULT. 2, AFTER_PERM_FAULT.

Type

int

Default value

0x0

auxilliary_feature_register0

Value of AFR0 ID register.

Type

int

Default value

0x0

branch-predictor-clear-policy

Set branch prediction policy as defined for MMFR1[31:28]. This does not change the behaviour of the branch predictor, only what is reported in MMFR1.BPred.

Type

int

Default value

0x2

branch-predictor-supported-ops

Set branch prediction policy as defined for MMFR3[11:8]. This does not change the behaviour of the branch predictor, only what is reported in MMFR3.BPMaint.

Type

int

Default value

0x1

brbe_log2_num_records

Log2 of number of BRB records supported. 3 -> 8 records, ... 6 -> 64 records.

Type

int

Default value

0x6

cache-log2linelen

Log2 of the cache line length in bytes.

Type

int

Default value

0x6

cache_maintenance_hits_watchpoints

DCIMVA operations executed in AArch32 modes hit watchpoints.

Type

bool

Default value

0x0

changing_block_size_without_bbm_support

Level of support for changing block size without break-before-make (FEAT_BBM).

Type

int

Default value

0x0

check_memory_attributes

Detect and report TLB use of conflicting memory attributes for views of the same physical address

Type

bool

Default value

0x1

checked_pointer_arithmetic_support_level

Specify the Checked Pointer Arithmetic support level: 0, not implemented. 1, FEAT_CPA is implemented. 2, FEAT_CPA2 is implemented.

Type

int

Default value

0x0

clean_invalidate_cache_on_warm_reset

Clean and invalidate caches on warm reset

Type

bool

Default value

0x0

clear_IT_when_IL_set

Clear IT bits when performing a *legal* exception return to AArch32 when IL is set

Type

bool

Default value

0x0

clear_IT_when_IL_set_explicitly

Apart from clear_IT_when_IL_set, also clear IT bits when loading CPSR from SPSR/memory and IL == 1 in the value being loaded.

Type

bool

Default value

0x0

clear_ec_in_debug_state

When ARMv8.8 debug extension is implemented, whether EDES.RC bit is set/cleared on entering debug state due to pending exception catch caused by EDES.RC=1.

Type

bool

Default value

0x0

clear_reg_top_eret

Behaviour of the upper 32-bits of the Xn registers when changing between AArch32 state and AArch64 state. 0, upper 32-bits preserved for all registers. 1, upper 32-bits set to 0 for all accessible registers. 2, upper 32-bits set to 0 for a random selection of accessible registers. 3, upper-32-bits set to 0 for registers touched in AArch32

Type

int

Default value

0x1

clear_reg_top_set

Whether to clear upper 32-bits of the Xn register when corresponding AArch32 register is set via CADI/Iris.

Type

bool

Default value

0x1

configure_pmu_events_with_json

"Configure v8.6 and newer PMU events. Note : This param has high priority and overrides the setting of "has_*_pmu_events" if both the params are provided. JSON schema for the parameter value is e.g. 1. {"all":false} 2. {"pmu_events":["EVENT_NAME_1","EVENT_NAME_2"]}"

Type

string

Default value**configure_v8_6_pmu_events_with_json**

"[DEPRECATED: Set configure_pmu_events_with_json to the same value instead] Configure v8.6 PMU events. Note : This param has high priority and overrides the setting of "has_v8_6_pmu_events" if both the params are provided. JSON schema for the parameter value is e.g. 1. {"all":false} 2. {"pmu_events":["BR_INDNR_RETIRE", "BR_IND_RETIRE",

"BR_RETURN_SKIP_RETIRED", "BR_RETURN_ANY_RETIRED", "BR_INDNR_SKIP_RETIRED",
 "BR_INDNR_TAKEN_RETIRED", "BR_IND_SKIP_RETIRED", "BR_IND_TAKEN_RETIRED",
 "BR_IMMED_SKIP_RETIRED", "BR_IMMED_TAKEN_RETIRED", "BR_SKIP_RETIRED"]}]"

Type

string

Default value**configure_v8_8_pmu_events_with_json**

"[DEPRECATED: Set configure_pmu_events_with_json to the same value instead] Configure v8.8 PMU events. Note : This param has high priority and overrides the setting of "has_v8_8_pmu_events" if both the params are provided. JSON schema for the parameter value is e.g. 1. {"all":false} 2. {"pmu_events":["BR_HINT_COND_RETIRED", "BR_COND_TAKEN_RETIRED", "BR_UNCOND_RETIRED", "BR_COND_RETIRED", "BRNL_TAKEN_RETIRED", "BRNL_IND_TAKEN_RETIRED", "BRNL_INDNR_TAKEN_RETIRED", "BRNL_IMMED_TAKEN_RETIRED", "BL_TAKEN_RETIRED", "BL_IND_TAKEN_RETIRED", "BL_IMMED_TAKEN_RETIRED"]}]"

Type

string

Default value**configure_v8_9_pmu_events_with_json**

"[DEPRECATED: Set configure_pmu_events_with_json to the same value instead] Configure v8.9 PMU events. Note : This param has high priority and overrides the setting of "has_v8_9_pmu_events" if both the params are provided. JSON schema for the parameter value is e.g. 1. {"all":false} 2. {"pmu_events":["ASE_SVE_RETIRED", "ASE_RETIRED", "VFP_RETIRED", "SVE_RETIRED", "CRYPTO_RETIRED", "SIMD_INST_RETIRED", "ASE_INST_RETIRED", "SVE_INST_RETIRED", "ASE_SVE_INST_RETIRED", "LD_ANY_RETIRED", "ST_ANY_RETIRED", "LDST_ANY_RETIRED", "DP_RETIRED"]}]"

Type

string

Default value**core_cache_protection**

core_cache_protection can change ERROFR, ERROPFGF and ERROPFGCTL fields. Possible values are: -1:Not implemented (by default), 0:Disabled, 1:Enabled

Type

int

Default value

0xffffffffffffffff

cpacr_trcds_behaviour

Behaviour of CPACR.TRCDIS/NSACR.NSTRCDIS when there is no CP14 ETM interface. 0, RAZ/WI. 2, implemented.

Type

int

Default value

0x2

cpu_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CONFIG64

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CP15SDISABLE2

Initialize to disable access to some CP15 registers (FEAT_CP15SDISABLE2).

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.DCZID-log2-block-size

Log2 of the block size cleared by DC ZVA instruction (as read from DCZID_EL0).

Type

int

Default value

0x8

cpuX.DCZVA_single_write

Execute the DCZVA as a single write

Type

bool

Default value

0x0

cpuX.MPIDR-override

Override of MPIDR value. If nonzero will override the MT, cluster and CPU ID bits in MPIDR.

Type

int

Default value

0x0

cpuX.RVBAR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.RVBAR32

Reset vector address in AARCH32 when VINITHI is not set and ignore_rvbar_in_aarch32 is set

Type

int

Default value

0x0

cpuX.SMPnAMP

Enable broadcast messages necessary for correct SMP operation at reset.

Type

bool

Default value

0x1

cpuX.TEINIT

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.aarch32_reset_from_impdef_addr

If PE resets into AArch32, Whether execution starts from IMPDEF address or hi/low vector

Type

bool

Default value

0x1

cpuX.ase-present

Set whether the model has been built with NEON support

Type

bool

Default value

0x1

cpuX.clock_divider

Clock divider ratio for asymmetric MP clocking.

Type

int

Default value

0x1

cpuX.clock_multiplier

Clock divider ratio for asymmetric MP clocking.

Type

int

Default value

0x1

cpuX.crypto_aes

AES instructions supported (requires CryptoPlugin to be loaded). 0, not implemented. 2, AES and PMULL instructions implemented (FEAT_AES, FEAT_PMULL).

Type

int

Default value

0x2

cpuX.crypto_sha1

SHA-1 instructions supported (requires CryptoPlugin to be loaded). 0, not implemented. 1, SHA1 instructions implemented (FEAT_SHA1).

Type

int

Default value

0x1

cpuX.crypto_sha256

SHA-256 instructions supported (requires CryptoPlugin to be loaded). 0, not implemented. 1, SHA256 instructions implemented (FEAT_SHA256).

Type

int

Default value

0x1

cpuX.crypto_sha3

Implement ARMv8.4 SHA-3 instructions (requires CryptoPlugin to be loaded) (FEAT_SHA3).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

cpuX.crypto_sha512

Implement ARMv8.4 SHA-512 instructions (requires CryptoPlugin to be loaded) (FEAT_SHA512).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

cpuX.crypto_sm3

Implement ARMv8.4 SM-3 instructions (requires CryptoPlugin to be loaded) (FEAT_SM3).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

cpuX.crypto_sm4

Implement ARMv8.4 SM-4 instructions (requires CryptoPlugin to be loaded) (FEAT_SM4).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

cpuX.cti-intack_mask

Set bits represent that the corresponding trigger requires software acknowledge via CTIINTACK

Type

int

Default value

0x1

cpuX.cti-number_of_claim_bits

Number of implemented bits in CTICLAIMSET

Type

int

Default value

0x0

cpuX.cti-number_of_triggers

Number of cti event triggers (default: 8, valid values: {3, 8-32})

Type

int

Default value

0x8

cpuX.enable_crc32

CRC32 instructions supported. 0, not implemented. 1, CRC32 instructions implemented (FEAT_CRC32).

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.etm-present

Set whether the model has ETM support

Type

bool

Default value

0x1

cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x0

cpuX.force-fpsid-value

Value to override the FPSID value to

Type

int

Default value

0x0

cpuX.has_hcptr_tase

If false, HCPtr.TASE is RES0

Type

bool

Default value

0x1

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.number-of-breakpoints

Number of breakpoints.

Type

int

Default value

0x10

cpuX.number-of-context-breakpoints

Number of breakpoints that are context aware.

Type

int

Default value

0x10

cpuX.number-of-watchpoints

Number of watchpoints.

Type

int

Default value

0x10

cpuX.operation_bandwidth

Operation width for ARMv8.4 PMU extension

Type

int

Default value

0x1

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.semihosting-stderr_istty

Result for semihost istty call when argument is stderr

Type

bool

Default value

0x1

cpuX.semihosting-stdin_istty

Result for semihost istry call when argument is stdin

Type

bool

Default value

0x1

cpuX.semihosting-stdout_istty

Result for semihost istry call when argument is stdout

Type

bool

Default value

0x1

cpuX.semihosting-use_stderr

Send stderr from the simulated process to host stderr

Type

bool

Default value

0x0

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.unpredictable_WPMASKANDBAS

Constrained unpredictable handling of watchpoints when mask and BAS fields specified. 0, IGNOREMASK. 1, IGNOREBAS (default). 2, REPEATBAS8. 3, REPEATBAS.

Type

int

Default value

0x1

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

cpuX.vfp-traps

Implement support for trapping floating-point exceptions

Type

bool

Default value

0x1

cpuX.vfp-traps-show-all

Report all trapped floating-point exceptions in the syndrome when a combination occurs.

Type

bool

Default value

0x0

cpuX.wfet_early_or_delayed_timeout

WFET early or delayed timeout beyond the threshold value of CNTVCT_ELO in percentage.

Type

int

Default value

0x0

cpuX.wfit_early_or_delayed_timeout

WFIT early or delayed timeout beyond the threshold value of CNTVCT_ELO in percentage.

Type

int

Default value

0x0

cpuselr_el3_sync_immediate

Adjust when the patching selection register synchronises - either immediately (true - default), or awaiting for barrier event

Type

bool

Default value

0x1

cpy_mops_option

Set option for Armv8.8 CPY(FEAT_MOPS). 0, use default(i.e. use value configured through has_mops_option). 1, implemented using Option A. 2, implemented using Option B

Type

int

Default value

0x0

cpyf_mops_option

Set option for Armv8.8 CPYF(FEAT_MOPS). 0, use default(i.e. use value configured through has_mops_option). 1, implemented using Option A. 2, implemented using Option B

Type

int

Default value

0x0

cycle_counter_freeze_on_spe_event

Whether pmu cycle counter does not count when pmcr_el0.dp=1 and pmu event counters are frozen by pmcr_el0.fzs.

Type

bool

Default value

0x0

d128_disabled_ps_resvd_size

Physical size treated when TCR.(I)PS is programmed with value seven and D128 is disabled via TCR. 0, 48 bits. 1, 52 bits. 2, 56.

Type

int

Default value

0x2

dbg-bcr-reserved-behavior

This is the behavior of the reserved values of the BT field in DBGBCR. Possible values are: - 0 = Disabled. - 1 = BT[2] is ignored.

Type

int

Default value

0x1

dbg_rom_dap_addr

Debug ROM dap base address.

Type

int

Default value

0x0

dbgitr_buffer_size

Number of instructions which can be buffered before EDSCR.ITE is cleared

Type

int

Default value

0x0

dbgxvr_ress_is_stateful

Whether DBGWVR/DBGBVR.RESS returns last written value. if set to false, RESS returns sign extended value

Type

bool

Default value

0x0

dc_fault_unaligned_s1_device_s2_fwb

Whether takes an Alignment Fault caused by the memory type on a DC {ZVA,GZVA,GVA} if the stage 1 memory type is any Device memory type

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_bus_width_in_bytes

L1 D-Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. `dcache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-ways

L1 D-Cache number of ways (sets are implicit from size).

Type

int

Default value

0x2

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_bus_width_in_bytes

L1 D-Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcimva_requires_s2_write_permissions

Data-cache invalidate by MVA operations require stage 2 write permission (virtualised AArch32 guest).

Type

bool

Default value

0x0

dczva_reports_lowest_addr_on_tag_check_fail

Whether DC ZVA reports lowest address in FAR on tag check fail

Type

bool

Default value

0x0

dczva_wp_far_behaviour

Set option for address stored in FAR/EDWARD after watchpoints hit by DC ZVA. 0 - FAR recorded matches lowest watchpointed address accessed by the instruction 1 - FAR recorded matches lowest address accessed by the instruction within same translation granule as watchpointed address 2 - FAR recorded matches highest address accessed by the instruction within same translation granule as watchpointed address

Type

int

Default value

0x0

debug_auth_signals_sampled_at_reset

Debug authentication signals can be configured as either sampled at reset only or at any time for External Root Debug.

Type

bool

Default value

0x0

debug_components_dap_address

Debug components ROM,ED,CTI,PMU,TRACE and TRBU base address for each core on debug bus. The "rom" field in the "cores" array are only allowed when 'debug_rom_is_flat' is false. JSON schema for the parameter value is: {"format": "all_addrs_are_absolute_wrt_debugbus", "cores": [{"rom": 0x0, "ed": 0x0, "cti": 0x0, "pmu": 0x0, "etm": 0x0, "trbu": 0x0}, {"rom": 0x0, "ed": 0x0, "cti": 0x0, "pmu": 0x0, "etm": 0x0, "trbu": 0x0}, {"rom": 0x0, "ed": 0x0, "cti": 0x0, "pmu": 0x0, "etm": 0x0, "trbu": 0x0}, {"rom": 0x0, "ed": 0x0, "cti": 0x0, "pmu": 0x0, "etm": 0x0, "trbu": 0x0}]}

Type

string

Default value**debug_components_mmap_address**

Debug components ROM,ED,CTI,PMU,TRACE and TRBU base address for each core on system bus. The "rom" field in the "cores" array are only allowed when 'debug_rom_is_flat' is false. JSON schema for the parameter value is: {"format": "all_addrs_are_absolute_wrt_systembus", "cores": [{"rom": 0x0, "ed": 0x0, "cti": 0x0, "pmu": 0x0, "etm": 0x0, "trbu": 0x0}, {"rom": 0x0, "ed": 0x0, "cti": 0x0, "pmu": 0x0, "etm": 0x0, "trbu": 0x0}, {"rom": 0x0, "ed": 0x0, "cti": 0x0, "pmu": 0x0, "etm": 0x0, "trbu": 0x0}, {"rom": 0x0, "ed": 0x0, "cti": 0x0, "pmu": 0x0, "etm": 0x0, "trbu": 0x0}]}

```
"trbu":0x0}, {"rom":0x0, "ed":0x0, "cti":0x0, "pmu":0x0, "etm":0x0, "trbu":0x0}, {"rom":0x0, "ed":0x0, "cti":0x0, "pmu":0x0, "etm":0x0, "trbu":0x0}}}
```

Type

string

Default value**debug_entry_is_context_sync**

If true, Entry in debug state is Context sync. Exiting debug state is a context synchronising operation, but entering is not. However some cpu implementation can consider also the Entry in Debug state as a CSE

Type

bool

Default value

0x0

debug_rom_is_class_9

If true, present a debug ROM table as a class 9 device. Otherwise, use a class 1 ROM table.

Type

bool

Default value

0x0

debug_rom_is_flat

If true, present a debug ROM table recommended by ARMv8 Debug Architecture. Otherwise, use nested ROM tables.

Type

bool

Default value

0x0

delay_serror

Add a propagation delay of serror signal into the core

Type

int

Default value

0x0

delayed_dbgreg_between_secure_views

If delayed_dbgreg is enabled, whether the secure and nonsecure external views require explicit synchronization.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

delayed_pmureg_between_secure_views

If delayed_pmureg is enabled, whether the secure and nonsecure external views require explicit synchronization.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

dfr1_reads_actual_bp_wp_ctx_cmp

If true, the register ID_AA64DFR1_EL1/EDDFR1 reports the actual number of BRPs, WRPs and CTX_CMPs even when the number of bp/wp/ctx_cmp is less than 16.

Type

bool

Default value

0x0

dic-spi_count

Number of shared peripheral interrupts implemented

Type

int

Default value

0x40

disable_sve_plugin

If true, SVE will not be implemented in this processor even if the plugin is loaded (FEAT_SVE).

Type

bool

Default value

0x0

disable_unknown_update_event_on_reset

Disables SYSREG_UPDATE event notification on reset for the registers whose bitfields are all reserved or resets to architecturally unknown value

Type

bool

Default value

0x0

dsb_accumulate_threshold

Limit the maximum number of observable DSB side effects which can be queued (e.g. TLBI), after which DSB sync will be done automatically.

Type

int

Default value

0x100

early_implicit_error_sync_event_behaviour

Set option for Early Implicit Error Synchronization event (FEAT_IESB) 0x0 - Behavior is not described ID_AA64MMFR4_EL1.EIESB = 0x0 0x1 - Implicit Error synchronization event is inserted before an exception is taken to EL3 (depending on SCR_EL3.NMEA) ID_AA64MMFR4_EL1.EIESB = 0x1 0x2 - Implicit Error synchronization event is inserted before an exception is taken to ELx (depending on SCR_EL3.NMEA and SCTLR2_ELx.NMEA) ID_AA64MMFR4_EL1.EIESB = 0x2 0xF - Implicit Error synchronization event is inserted after an exception is taken ID_AA64MMFR4_EL1.EIESB = 0xF

Type

int

Default value

0x0

ecv_support_level

Implement Enhanced Counter Virtualization feature from ARMv8.6. 0, Not supported. 1, fully supported without CNTPOFF. 2, fully supported with CNTPOFF (FEAT_ECV).

Type

int

Default value

0x0

eddfr1_reads_idreg_mask

Mask to configure each bitfield for EDDFR1 register, whether to be read from corresponding bitfield in AA64DFR1 register. Note: Only those bitfield which supports 'same as ID register or res0' functionality can be configured

Type

int

Default value

0xffffffffffffffff

eddfr2_reads_idreg_mask

Mask to configure each bitfield for EDDFR2 register, whether to be read from corresponding bitfield in ID_AA64DFR2 register. Note: Only those bitfield which supports 'same as ID register or res0' functionality can be configured

Type

int

Default value

0xffffffffffffffff

eddfr_reads_idreg_mask

Mask to configure each bitfield for EDDFR register, whether to be read from corresponding bitfield in AA64DFR register. Note: Only those bitfield which supports 'same as ID register or res0' functionality can be configured

Type

int

Default value

0xffffffffffffffff

edpfr_ras_unknown_bits_read_as_0

If true then UNKNOWN bits in RAS field in EDPFR are read as 0

Type

bool

Default value

0x0

edxfr_reads_idreg

Whether EDDFR,EDDFR1 and EDDFR2 reads corresponding bitfield value from ID_AA64DFR reg. Also, when this parameter is enabled, bitfields of these registers are configurable through 'eddfr*_reads_idreg_mask' parameter

Type

bool

Default value

0x0

e10_can_access_imp_def_functionality

If not made UNDEF by imp_def_functionality_behaviour, ELO can access IMPLEMENTATION DEFINED registers and system instructions.

Type

bool

Default value

0x0

e10_e11_only_non_secure

Secure/non-secure state if EL2 and EL3 are not implemented. 0, secure. 1, non-secure.

Type

bool

Default value

0x0

e13_trap_priority_when_secure_debug_disabled

Undef when secure debug is disabled (EDSCR.SDD == 1) && boolean IMPLEMENTATION_DEFINED 'EL3 trap priority when SDD == 1'

Type

bool

Default value

0x0

enable_address_contig_check

Check the input address range for the table entries that have the contiguous hint bit set.

Type

bool

Default value

0x0

enable_debug_auth_signals_config

Debug Authentication Signals DBGEN, SPIDEN (and if RME is enabled RLPIDEN and RTPIDEN) are configurable (default) or not configurable, (hardwired to 1). This parameter is the integer representation of a bitmap to enable configuration of these signals, with: - BIT[0] = DBGEN - BIT[1] = SPIDEN - BIT[2] = RLPIDEN - BIT[3] = RTPIDEN

Type

int

Default value

0xf

enable_tlb_contig_check

Perform extra pagetable walks to check translation table entries that have the contiguous hint bit set.

Type

bool

Default value

0x0

enhanced_pac2_level

Implements Enhanced PAC2 from ARMv8.6 (FEAT_PAuth2), and PAC enhancements from ARMv9.5 (FEAT_PAuth_LR). options 0-3 of this feature are mandatory for ARMv8.6 but can be cherry-picked to a ARMv8.3(or greater) implementation. 0: No EnhancedPAC2, 1: EnhancedPAC2 Only (FEAT_PAuth2), 2: EnhancedPAC2 with FPAC (FEAT_FPAC), 3: EnhancedPAC2 with FPACCombined (FEAT_FPACCOMBINE), 4: EnhancedPAC2 with LR signing (FEAT_PAuth_LR).

Type

int

Default value

0x0

error_record_feature_register

RAS feature register values. An array of JSON objects. The JSON schema for the array is: [{"ED":0x0, "IMPDEF_3_2":0x0, "UI":0x0, "FI":0x0, "UE":0x0, "CFI":0x0, "CEC":0x0, "RP":0x0, "DUI":0x0, "CEO":0x0, "CI":0x0, "TS":0x0, "INJ":0x0, "FRX":0x0, "UC":0x0, "UEU":0x0, "UER":0x0, "UEO":0x0, "DE":0x0, "CE":0x0, "Visibility":"Core"},other_feature_register_values]. Where ED,UI,FI,CE and UE have valid values between 0x0 - 0x3. CFI and DUI have valid values 0x0, 0x2 and 0x3. CEC has valid values 0x0,0x2 or 0x4. RP,CEO,INJ,FRX,UC,UEU,UER,UEO,DE has valid values 0x0 or 0x1. CI and TS has valid values of 0x0, 0x1 and 0x2. Visibility has valid values "Core" or "Cluster"

Type

string

Default value**error_record_feature_register_json_file**

File path to the RAS feature register values as JSON. The file uses the same format as the error_record_feature_register parameter value

Type

string

Default value**erxpfctl_res0_stateful_mask**

Mask for stateful bits for ERXPFGCTL which are RES0

Type

int

Default value

0x0

esr_write_update_res0

If true, and RASv2 is enabled, then ESR_ELx.WU field is RES0 for errors on both loads and stores (FEAT_RASv2).

Type

bool

Default value

0x0

ete.ASYNC_PACKETS_WHEN_VIEWINST_OFF

Generate the non-periodic alignment synchronisation packet generation when trace unit is operative

Type

bool

Default value

0x0

ete.ATBTRIG

ATB trigger support

Type

bool

Default value

0x1

ete.CCITMIN

Minimum cycle count value

Type

int

Default value

0x4

ete.CCSIZE

Cycle counter size

Type

int

Default value

0xc

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x8

ete.COMMOPT

Commit mode

Type

bool

Default value

0x1

ete.COMMTRANS

Commit transaction mode

Type

bool

Default value

0x0

ete.DEBUG

DEBUG

Type

int

Default value

0x2

ete.DESIGNER

DESIGNER value

Type

int

Default value

0x41

ete.ETE_REVISION

ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3

Type

int

Default value

0x0

ete.EXCEPTION_WITH_CONTEXT

Whether EXCEPTION_WITH_CONTEXT packet is supported

Type

bool

Default value

0x1

ete.EXPLICITLY_COMMIT_PO_ELEMS

Whether to unilaterally explicitly emit a commit after a PO packet

Type

bool

Default value

0x0

ete.IMPDEFEXCEPPERCENTAGE

Percentage of IMPDEF exceptions inserted in instruction blocks

Type

int

Default value

0x0

ete.IMPDEF_TRACE_ONWhether trace is flushed and trace on packet generated by events described by bitmap value.
bit 0 - PE entering low power state, bit 1 - PE entering debug state**Type**

int

Default value

0x0

ete.IMPRECISE_FILTERING

Number of instruction blocks traced on a transition in the filtering

Type

int

Default value

0x0

ete.LPOVERRIDE

Low power override

Type

bool

Default value

0x1

ete.MAXSPEC

Maximum speculation depth

Type

int

Default value

0x0

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NOOVERFLOW

No overflow

Type

bool

Default value

0x0

ete.NUMACPAIRS

Number of instruction address comparators pairs

Type

int

Default value

0x4

ete.NUMCIDC

Number of context ID comparators

Type

int

Default value

0x1

ete.NUMCNTR

Number of counters

Type

int

Default value

0x2

ete.NUMEXTINSEL

Number of external input selectors

Type

int

Default value

0x4

ete.NUMPC

Number of PE comparators

Type

int

Default value

0x0

ete.NUMSEQSTATE

Number of sequencer states

Type

int

Default value

0x4

ete.NUMSSCC

Number of single shot comparators

Type

int

Default value

0x1

ete.NUMVMIDC

Number of virtual ID comparators

Type

int

Default value

0x1

ete.NumberOfETEEEvents

Number of trace events

Type

int

Default value

0x2

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_DESIGNER

TRCPIDR DESIGNER value

Type

int

Default value

0x0

ete.PIDR_PART

TRCPIDR PART number value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.QFILT

Q filtering

Type

bool

Default value

0x0

ete.QSUP

Q support

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RAZWI_REG_SEL_TOP_BIT

Implement Resource Selectors or Resource Selector Pairs bits as RAZ/WI

Type

bool

Default value

0x0

ete.REG_ACCESS_ONLY_MODE

If enabled, all traces are disabled. Plugin only allows register acceses

Type

bool

Default value

0x0

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.STALLCTRL

Stall control

Type

bool

Default value

0x1

ete.SYSSTALL

System stall

Type

bool

Default value

0x1

ete.TRACEIDSIZE

Trace ID size

Type

int

Default value

0x7

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRACE_OUTPUT_ENABLE**

ETE Trace output enable: 1=enable, 0=disable

Type

bool

Default value

0x0

ete.TRCSRSTA_FORCED_EXCEP

TRCSRSTA value for a forcibly traced exception

Type

bool

Default value

0x0

ete.TSMARK

Whether timestamp markers are supported

Type

bool

Default value

0x0

ete.TSSIZE

Timestamp size

Type

int

Default value

0x8

ete.WFXMODE

WFX mode

Type

bool

Default value

0x1

ets_level

Level of Enhanced Translation Synchronization supported. 0: not supported (may be override to 1 if has_ets=true), 1: supported FEAT_ETTS, 2: supported FEAT_ETTS2

Type

int

Default value

0x0

exception_catch_before_software_step

Exception catch priority for the exception trapping form of exception catch (Armv8.2 or later, or exception_catch_type=0). If true, the exception catch debug event has higher priority than software step and halting step.

Type

bool

Default value

0x1

exception_catch_type

Type of exception catch (Armv8.0 - Armv8.1 only). 0, exception trapping. 1, non-exception trapping, higher priority than step. 2, non-exception trapping, lower priority than step.

Type

int

Default value

0x0

exclusive_monitor_clear_on_atomic_from_same_master

Exclusive monitors in the cluster will be cleared by a atomic by the same master to the monitored address.

Type

bool

Default value

0x1

exclusive_monitor_clear_on_store_from_same_master

Exclusive monitors in the cluster will be cleared by a store by the same master to the monitored address.

Type

bool

Default value

0x1

exclusive_monitor_clear_on_strex_address_mismatch

Exclusive monitors in the cluster will be cleared when a strex fails because the address does not match.

Type

bool

Default value

0x1

exclusive_monitor_clear_on_strex_success

Exclusive monitors in the cluster will be cleared when a strex succeeds.

Type

bool

Default value

0x1

exercise_stxr_fail

Reject a pseudo-random majority of exclusive store instructions

Type

bool

Default value

0x0

ext_abort_device_GRE_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffffffff

ext_abort_device_GRE_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffffffff

ext_abort_device_GRE_read_is_critical

Critical reporting of device-GRE read external aborts.

Type

bool

Default value

0x0

ext_abort_device_GRE_read_is_sync

Synchronous reporting of device-GRE read external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_device_read_is_sync.

Type

int

Default value

0x2

ext_abort_device_GRE_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_device_read_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffffffff

ext_abort_device_GRE_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_device_read_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffffffff

ext_abort_device_GRE_write_is_critical

Critical reporting of device-GRE write external aborts

Type

bool

Default value

0x0

ext_abort_device_GRE_write_is_sync

Synchronous reporting of device-GRE write external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_device_write_is_sync.

Type

int

Default value

0x2

ext_abort_device_GRE_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_device_write_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_GRE_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_device_write_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_read_is_critical

Critical reporting of device-nGRE read external aborts.

Type

bool

Default value

0x0

ext_abort_device_nGRE_read_is_sync

Synchronous reporting of device-nGRE read external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_device_read_is_sync.

Type

int

Default value

0x2

ext_abort_device_nGRE_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_device_read_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_device_read_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_write_is_critical

Critical reporting of device-nGRE write external aborts

Type

bool

Default value

0x0

ext_abort_device_nGRE_write_is_sync

Synchronous reporting of device-nGRE write external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_device_write_is_sync.

Type

int

Default value

0x2

ext_abort_device_nGRE_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_device_write_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_nGRE_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_device_write_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_device_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_device_read_acquire_is_sync

Synchronous reporting of device read with acquire external aborts

Type

bool

Default value

0x0

ext_abort_device_read_is_critical

Critical reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x1

ext_abort_device_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_device_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_device_write_is_critical

Critical reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_device_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_device_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_fill_data

Returned data, if external aborts are asynchronous

Type

int

Default value

0xfdfdfdfcfcfdfdfd

ext_abort_normal_cacheable_read_is_critical

Critical reporting of normal write-back cacheable-read external aborts

Type

bool

Default value

0x0

ext_abort_normal_cacheable_read_is_sync

Synchronous reporting of normal write-back cacheable-read external aborts

Type

bool

Default value

0x1

ext_abort_normal_cacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_normal_cacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_normal_cacheable_write_is_critical

Critical reporting of normal write-back cacheable write external aborts

Type

bool

Default value

0x0

ext_abort_normal_cacheable_write_is_sync

Synchronous reporting of normal write-back cacheable write external aborts

Type

bool

Default value

0x0

ext_abort_normal_cacheable_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_normal_cacheable_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_normal_noncacheable_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_noncacheable_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_noncacheable_read_is_critical

Critical reporting of normal noncacheable-read external aborts

Type

bool

Default value

0x0

ext_abort_normal_noncacheable_read_is_sync

Synchronous reporting of normal noncacheable-read external aborts

Type

bool

Default value

0x1

ext_abort_normal_noncacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_normal_noncacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_normal_noncacheable_write_is_critical

Critical reporting of normal noncacheable write external aborts

Type

bool

Default value

0x0

ext_abort_normal_noncacheable_write_is_sync

Synchronous reporting of normal noncacheable write external aborts

Type

bool

Default value

0x0

ext_abort_normal_noncacheable_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_normal_noncacheable_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_normal_wt_cacheable_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_wt_cacheable_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

ext_abort_normal_wt_cacheable_read_is_critical

Critical reporting of normal write-through cacheable-read external aborts

Type

bool

Default value

0x0

ext_abort_normal_wt_cacheable_read_is_sync

Synchronous reporting of normal write-through read external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_normal_cacheable_read_is_sync.

Type

int

Default value

0x2

ext_abort_normal_wt_cacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_normal_cacheable_read_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffffffff

ext_abort_normal_wt_cacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_normal_cacheable_read_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffffffff

ext_abort_normal_wt_cacheable_write_is_critical

Critical reporting of normal write-through write external aborts

Type

bool

Default value

0x0

ext_abort_normal_wt_cacheable_write_is_sync

Synchronous reporting of normal write-through write external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_normal_cacheable_write_is_sync.

Type

int

Default value

0x2

ext_abort_normal_wt_cacheable_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_normal_cacheable_write_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffffffff

ext_abort_normal_wt_cacheable_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_normal_cacheable_write_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffffffff

ext_abort_prefetch_device_GRE_read_is_critical

Critical reporting of external aborts generated by device-GRE instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_device_GRE_read_is_sync

Behaviour of external aborts generated by device-GRE instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_prefetch_device_nGRE_read_is_critical

Critical reporting of external aborts generated by device-nGRE instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_device_nGRE_read_is_sync

Behaviour of external aborts generated by device-nGRE instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_prefetch_device_read_is_critical

Critical reporting of external aborts generated by device-nGnRE instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_device_read_is_sync

Behaviour of external aborts generated by device-nGnRE instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_prefetch_is_critical

Critical reporting of external aborts generated by normal writeback cacheable instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_is_sync

Behaviour of external aborts generated by normal writeback cacheable instruction fetches. 0, asynchronous abort. 1, synchronous abort.

Type

bool

Default value

0x1

ext_abort_prefetch_noncacheable_read_is_critical

Critical reporting of external aborts generated by normal noncacheable instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_noncacheable_read_is_sync

Behaviour of external aborts generated by normal noncacheable instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_prefetch_so_read_is_critical

Critical reporting of external aborts generated by device-nGnRnE instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_so_read_is_sync

Behaviour of external aborts generated by device=nGnRnE instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_prefetch_wt_cacheable_read_is_critical

Critical reporting of external aborts generated by normal writethrough cacheable instruction fetches

Type

bool

Default value

0x0

ext_abort_prefetch_wt_cacheable_read_is_sync

Behaviour of external aborts generated by normal writethrough cacheable instruction fetches. 0, asynchronous abort. 1, synchronous abort. 2, same as ext_abort_prefetch_is_sync.

Type

int

Default value

0x2

ext_abort_so_prefetch_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffffffff

ext_abort_so_prefetch_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_prefetch_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffffffff

ext_abort_so_read_is_critical

Critical reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x1

ext_abort_so_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_so_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_so_write_is_critical

Critical reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x1

ext_abort_so_write_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_ttw_cacheable_read_is_critical

Critical reporting of TTW cacheable read external aborts

Type

bool

Default value

0x0

ext_abort_ttw_cacheable_read_is_sync

Synchronous reporting of TTW cacheable read external aborts

Type

bool

Default value

0x1

ext_abort_ttw_cacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_ttw_cacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_ttw_noncacheable_read_is_critical

Critical reporting of TTW noncacheable read external aborts

Type

bool

Default value

0x0

ext_abort_ttw_noncacheable_read_is_sync

Synchronous reporting of TTW noncacheable read external aborts

Type

bool

Default value

0x1

ext_abort_ttw_noncacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: Valid indices in range [0, number_of_error_records).

Type

int

Default value

0x0

ext_abort_ttw_noncacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_ttw_wt_cacheable_read_is_critical

Critical reporting of TTW write-through cacheable read external aborts

Type

bool

Default value

0x0

ext_abort_ttw_wt_cacheable_read_is_sync

Synchronous reporting of TTW write-through cacheable read external aborts. 0, asynchronous. 1, synchronous. 2, same as ext_abort_ttw_cacheable_read_is_sync.

Type

int

Default value

0x2

ext_abort_ttw_wt_cacheable_read_ras_index

External Aborts are reported in RAS record index specified in this param. Values: -1 = Same as ext_abort_ttw_cacheable_read_ras_index, Valid indices in range [0, number_of_error_records).

Type

int

Default value

0xffffffffffffffff

ext_abort_ttw_wt_cacheable_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: -1 = Same as ext_abort_ttw_cacheable_read_ras_type, 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0xffffffffffffffff

external_debug_request_delay

Configure External Debug Request delay in CPU cycles

Type

int

Default value

0x0

external_oslar_access_disabled_by_authentication

If true, external accesses to OSLAR, when external debugging is not enabled, will generate an error (FEAT_Debugv8p2).

Type

bool

Default value

0x0

fault_on_misprogrammed_gpt_contig_region

Whether GPF faults occur when GPT contiguous entries are misprogrammed.

Type

bool

Default value

0x0

fault_on_nT_bit_set

Whether block translation table entries with the nT bit set should always fault. Only applies when changing_block_size_without_bbm_support_level is 1 or higher.

Type

bool

Default value

0x1

fault_unalign_to_unsupported_access

If has_unaligned_single_copy_atomicity is true, whether unaligned A64 atomic, exclusive and acquire/release instructions to non iWB-oWB or the access crossing a 16-byte boundary

generate fault. Bits 0,1,2,3 should be set accordingly to enable the fault behaviour. bit 0: atomic access should fault, bit 1: exclusive access should fault, bit 2: acquire/release should fault, bit 3: the access crossing a 16-byte boundary should fault

Type

int

Default value

0x8

fault_unaligned_s1_device_s2_fwb

Whether unaligned fault with stage1 Device memory and final memory attribute forced to normal by FWB. 0, No fault. 1, will fault. 2 No fault if final Shareability is NSH.

Type

int

Default value

0x0

force_align_pc

UNPREDICTABLE branch to non-word-aligned address in ARM state is forced to be aligned

Type

bool

Default value

0x0

force_deterministic_irg_tag_generation

Force the random tag generated by the IRG instruction when GCR_EL1.RRND=1 to equal RGSR_EL1.SEED[3:0] rather than a non-deterministic value

Type

bool

Default value

0x0

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_wnr_read_unsupported_exclusive_or_atomic

Whether ESR_ELx.WnR is forced to 0 for unsupported atomic and exclusives

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

fp8_support_level

0->No support for Advanced SIMD, SVE2 FP8 instructions 1->Support for FEAT_FP8
2->Support for FEAT_FP8FMA 3->Support for FEAT_FP8DOT4 4->Support for FEAT_FP8DOT2

Type

int

Default value

0x0

fpcr_short_vector_raz

FPSCR and FPCR fields LEN and STRIDE are hardwired to 0

Type

bool

Default value

0x0

fpsr_res0_stateful_mask

Mask for stateful bits of FPSR which are RES0

Type

int

Default value

0x0

fsr_ext_bit_update_kind

Set/Clear DFSR/IFSR EA bit on Synchronous/Async External Aborts. 0: Never Set, 1: Set on Synchronous Ext Aborts 2: Set on Asynchronous Ext Aborts 3: Set on both Sync and Async Ext Aborts.

Type

int

Default value

0x3

gcs_data_check_overrides_data_abort

GCS Data check exceptions are taken before Data Aborts

Type

bool

Default value

0x0

gcs_overshoot_writes

Number of overshooting GCS records written after a writing a record

Type

int

Default value

0x0

gic.GICC-offset

Offset from PERIPHBASE for GICC registers.

Type

int

Default value

0x2000

gic.GICD-offset

Offset from PERIPHBASE for GICD registers. Will be ignored when GICv3 CPU interface is enabled, as distributor is then external to the cluster.

Type

int

Default value

0x1000

gic.GICH-offset

Offset from PERIPHBASE for GICH registers.

Type

int

Default value

0x4000

gic.GICH-other-CPU-offset

Offset from PERIPHBASE for GICH registers for accessing other CPUs in the cluster. Set to 0 to disable.

Type

int

Default value

0x5000

gic.GICV-alias

Offset from PERIPHBASE for alias of GICV registers. When gicv2-only, if zero no alias will be created; if gicv2-only=0, the param is deprecated, when zero or unset an alias is created in the place mandated by the architecture (GICV-base+0xF000)

Type

int

Default value

0x0

gic.GICV-offset

Offset from PERIPHBASE for GICV registers.

Type

int

Default value

0x6000

gic.PERIPH-size

Size of registers based at PERIPHBASE that are considered to be owned by the GIC. Any accesses in the range PERIPHBASE to PERIPHBASE+gic.PERIPH-size-1 that do not match GIC registers will be treated as RAZ/WI.

Type

int

Default value

0x8000

gicv3.A3-affinity-supported

Whether a non-zero value for affinity at level 3 is supported.

Type

bool

Default value

0x0

gicv3.BPR-min

The minimum value for the GICC_BPR register (non-secure version will be 1 + this value).

Type

int

Default value

0x2

gicv3.EOI-check-CPUID

Check CPU ID specified for accesses to EOI registers (rather than just ending highest priority active interrupt).

Type

bool

Default value

0x0

gicv3.EOI-check-ID

Check Interrupt ID specified for accesses to EOI registers (rather than just ending highest priority active interrupt).

Type

bool

Default value

0x0

gicv3.EOI-deactivate-any-interrupt

Allow an EOI to deactivate interrupts that aren't the highest priority active interrupt (EOI-ignore-out-of-order must be false otherwise this is ignored).

Type

bool

Default value

0x0

gicv3.EOI-ignore-out-of-order

Ignore EOI writes that cannot end the highest priority active interrupt.

Type

bool

Default value

0x1

gicv3.FIQEn-RAO

GICC_CTLR.FIQEn is read as one, write insensitive

Type

bool

Default value

0x0

gicv3.IIDR_base

The base value for calculating the GICC_IIDR register value.

Type

int

Default value

0x43b

gicv3.LR-count

The number of implemented list registers.

Type

int

Default value

0x10

gicv3.PMHE-RAO-WI

ICC_CTLR_EL*.PHME is read as one, write insensitive

Type

bool

Default value

0x0

gicv3.PMHE-RAZ-WI

ICC_CTLR_EL*.PHME is read as zero, write insensitive

Type

bool

Default value

0x0

gicv3.PMHE-release-set-packet

if PHME is enabled, whether a SET packet is released by CPU Intf in Upstream Ack window.

Type

bool

Default value

0x0

gicv3.SRE-EL2-enable-RAO

When ICC_SRE_EL2.SRE is RAO/WI, makes ICC_SRE_EL2.Enable RAO/WI

Type

bool

Default value

0x0

gicv3.SRE-EL3-enable-RAO

When ICC_SRE_EL3.SRE is RAO/WI, makes ICC_SRE_EL3.Enable RAO/WI

Type

bool

Default value

0x0

gicv3.SRE-EL3-set-once

Restrict SRE EL3 to be set only once

Type

bool

Default value

0x0

gicv3.SRE-enable-action-on-mmap

Allowed values are: 0-SRE one allows mmap access. 1-SRE one disables mmap access. 2-SRE one makes mmap access RAZ-WI.

Type

int

Default value

0x0

gicv3.STATUSR-implemented

If GICv3 CPU interface is being used, this determines whether the STATUS registers are implemented

Type

bool

Default value

0x1

gicv3.VBPR-min

The minimum value for the GICV_BPR register (non-secure version will be 1 + this value).

Type

int

Default value

0x2

gicv3.VFIQEn-RAO

ICH_VMCR_EL2.VFIQEn is read as one, write insensitive

Type

bool

Default value

0x0

gicv3.cpuintf-mmap-access-level

Allowed values are: 0-mmap access is supported for GICC,GICH,GICV registers. 1-mmap access is supported only for GICV registers. 2-mmap access is not supported.

Type

int

Default value

0x0

gicv3.dir-trap-support

The cpu supports separate trapping of ICC_DIR_EL1 to EL2

Type

bool

Default value

0x1

gicv3.el3_trap_priority_when_secure_debug_disabled

Undef to access priorities group register when secure debug is disabled

Type

bool

Default value

0x0

gicv3.extended-interrupt-range-support

Device has support for extended SPI/PPI ID ranges

Type

bool

Default value

0x0

gicv3.gicv2-only

Limit the GIC implementation to GICv2 features only

Type

bool

Default value

0x0

gicv3.idle-is-ff

For GICC/GICV RPR, when idle, return FF when true, minimum supported priority otherwise

Type

bool

Default value

0x1

gicv3.ignore-DIR-write-when-EOImode-not-set

Ignore UNPREDICTABLE access to GICC_DIR register.

Type

bool

Default value

0x1

gicv3.interrupt-bypass-support

Interrupt bypass support, set to false for devices not supporting interrupt bypass

Type

bool

Default value

0x1

gicv3.local-SEIs

Generate SEI to signal internal issues

Type

bool

Default value

0x0

gicv3.local-VSEIs

Generate VSEI to signal internal issues

Type

bool

Default value

0x0

gicv3.physical-ID-bits

Number of physical ID bits implemented.

Type

int

Default value

0x10

gicv3.priority-bits

Number of priority bits implemented.

Type

int

Default value

0x5

gicv3.send-PMHE-command-only-when-priority-changes

Send PMHE upstream command to distributor only when write to ICC_PMR_EL1 changes the priority

Type

bool

Default value

0x0

gicv3.sgi-range-selector-support

Device has support for the Range Selector feature for SGI

Type

bool

Default value

0x0

gicv3.suppress-virtual-enables-comms

In GICv3 only mode, prevents the GIC CPUIF from communicating UpstreamWrite/VirtualEnables to the IRI

Type

bool

Default value

0x1

gicv3.virtual-ID-bits

Number of virtual ID bits implemented.

Type

int

Default value

0x10

gicv3.virtual-lpi-support

When GICv3 is supported, indicates a cut down CPUIF interface with no support of VLPI (GICv3 only) when false

Type

bool

Default value

0x1

gicv3.virtual-priority-bits

Number of virtual priority bits implemented.

Type

int

Default value

0x5

gicv3.without-DS-support

GICv3 CPU interfaces do not support disabling security in the distributor (GICD_CTLR.DS=1)

Type

bool

Default value

0x0

gicv4.mask-virtual-interrupt

If true, virtual interrupts can be masked from being reported to virtual CPU interface by setting ICH_HCR_EL2.DVIM 1. No control otherwise.

Type

bool

Default value

0x0

global_debug_rom.ROMDEVID

Value of Debug Rom Device Identification Register

Type

int

Default value

0x0

global_debug_rom.ROMPIDR

Value of Debug Rom Peripheral Identification Register

Type

int

Default value

0x4000bb000

global_debug_rom.ROMPRIDR0

Value of Debug ROM Power RequestID Register

Type

int

Default value

0x1

gpccr_el3_gpcp_behaviour

Used to control impdef behaviour when GPCP=1 (0->Faults are always generated and reported, 1->Faults are not generated and reported), 2->Faults are generated and reported only for Arm recommended cases

Type

int

Default value

0x2

gpt_tlb_size

Number of separate GPT TLB entries.

Type

int

Default value

0x0

gpt_walkcache_size

Number of GPT walk cache entries.

Type

int

Default value

0x0

hardware_translation_table_update_implemented

Implement hardware translation table updates from ARMv8.1 (FEAT_HAFDBS).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.1 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has-gicv4.1

GICv4.1 is enabled, and all the features with GICv4.1 are implemented (FEAT_GICv4p1).

Type

bool

Default value

0x0

has_128_bit_atomic_instructions

Implement 128-bit Atomic Instructions (FEAT_LSE128)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_128_bit_tt_descriptors

Implement 128-bit Translation Table Descriptors (FEAT_D128)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_16bit_asids

Enable 16-bit ASIDs.

Type

bool

Default value

0x1

has_16bit_vmids

Implement support for 16-bit VMIDs from ARMv8.1 (FEAT_VMID16).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.1 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_16k_granule

Implement the 16k LPAE translation granule.

Type

bool

Default value

0x0

has_4k_granule

Implement the 4k LPAE translation granule.

Type

bool

Default value

0x1

has_52bit_address_with_16k

Implements Armv8.7 52-bit IPA/PA support for 16k (FEAT_LPA2).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_52bit_address_with_4k

Implements Armv8.7 52-bit IPA/PA support for 4k (FEAT_LPA2).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_56_bit_va

56-bit Physical Address, identified as (FEAT_LVA3)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_64bit_pmu_ext_access

Implement 64-bit pmu external interface access

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_64k_granule

Implement the 64k LPAE translation granule.

Type

bool

Default value

0x1

has_aarch32_dbgdidr_etc

DBGDIDR, DBGDRAR, DBGDSAR exist even if EL1 doesn't implement AArch32

Type

bool

Default value

0x1

has_aarch32_hpd

If true then hierarchical permission disable is supported in AArch32 (FEAT_AA32HPD)

Type

bool

Default value

0x0

has_aarch64

All implemented exception levels can run in AArch64

Type

bool

Default value

0x1

has_actlr2

If true ACTLR2 exists and ACTLR2(NS) is aliased to ACTLR_EL1[63:32]

Type

bool

Default value

0x0

has_actlr_virtualisation

If true ACTLR_EL12 is implemented and ACTLR_EL1 supports virtualisation

Type

bool

Default value

0x0

has_address_breakpoint_linking

Implement Address Breakpoint Linking Extension (FEAT_ABLE)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_amu

Implement activity monitor functionality from ARMv8.4 (FEAT_AMUv1).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_amu_ext64

Implement 64-bit external interface to the Activity Monitors (FEAT_AMU_EXT64).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_arm_v8-1

Implement the ARMv8.1 Extension.

Type

bool

Default value

0x0

has_arm_v8-2

Implement the ARMv8.2 Extension.

Type

bool

Default value

0x0

has_arm_v8-3

Implement the ARMv8.3 Extension.

Type

bool

Default value

0x0

has_arm_v8-4

Implement the ARMv8.4 Extension.

Type

bool

Default value

0x0

has_arm_v8-5

Implement the ARMv8.5 Extension.

Type

bool

Default value

0x0

has_arm_v8-6

Implement the ARMv8.6 Extension.

Type

bool

Default value

0x0

has_arm_v8-7

Implement the Armv8.7 Extension.

Type

bool

Default value

0x0

has_arm_v8-8

Implement the ARMv8.8 Extension.

Type

bool

Default value

0x0

has_arm_v8-9

Implement the ARMv8.9 Extension.

Type

bool

Default value

0x0

has_arm_v9-0

Implement the ARMv9.0 Extension.

Type

bool

Default value

0x0

has_arm_v9-1

Implement the ARMv9.1 Extension.

Type

bool

Default value

0x0

has_arm_v9-2

Implement the ARMv9.2 Extension.

Type

bool

Default value

0x0

has_arm_v9-3

Implement the ARMv9.3 Extension.

Type

bool

Default value

0x0

has_arm_v9-4

Implement the ARMv9.4 Extension.

Type

bool

Default value

0x0

has_arm_v9-5

Implement the ARMv9.5 Extension.

Type

bool

Default value

0x0

has_asid2

If true then support for use of two concurrent ASIDs (FEAT_ASID2)

Possible values of this parameter are: - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_at_with_pan

Implement new AT instructions with PAN support (FEAT_PAN2).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_ats1a

Support for ATS1ExR instructions (FEAT_ATS1A)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_attribute_index_enhancement

Memory Attribute Index Enhancement (FEAT_AIE)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_axflag_xaflag

Implement flag manipulation instructions (AXFlag, XAFlag) from ARMv8.5 (FEAT_FlagM2).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_axflag_xaflag_frint

Implement flag manipulation instructions (AXFlag, XAFlag) and floating-point rounding to int instructions (FRINT[32|64][X|Z]) from ARMv8.5. If this parameter is enabled, it also enables both has_axflag_xaflag and has_frint. If support for only one of the features is needed, please use the individual parameters and do not enable this one (FEAT_FlagM2, FEAT_FRINTTS).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_bc

Implement Armv8.8 Hinted Conditional Branch (FEAT_HBC)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_branch_target_exception

Implement Branch target identification mechanism from ARMv8.5 (FEAT_BTI).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_brbe

If true, implements branch record buffer extension (FEAT_BRBE).

Type

bool

Default value

0x0

has_brbe_v1p1

If true, implements FEAT_BRBEv1p1.

Type

bool

Default value

0x0

has_ccidx

Implement the ARMv8.3 CCSIDR Extension. Extending the ccidr number of sets.

Type

bool

Default value

0x0

has_cfinv_rmif_setf

Implement flag manipulation (CFINV, RMIF, SETF8, SETF16) instructions from ARMv8.4 (FEAT_FlagM).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_chkfeat

Implement CHKFEAT instruction from ARMv9.4 (FEAT_CHK).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_clear_bhb

Implement Clear Branch History information instruction (FEAT_CLRBHB).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_clear_other_speculation_by_context

Implement execution and data prediction invalidation from Armv8.9 (FEAT_SPECRES2).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_cmo_wr_control

Whether stage1/2 CMO write perm control is supported (FEAT_CMOW)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x0

has_common_not_private_translations

Implement the TTBRn_ELx.CnP (Common not Private) controls from ARMv8.2 (FEAT_TTCNP).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_complex_number

Implement ARMv8.3 complex number support, Multiply Accumulate and Add instructions (FEAT_FCMA).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.3 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_const_pac

Feature for singular selection of PAC field (FEAT_CONSTPACFIELD).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_cssc

Support for common short sequence compression instructions (FEAT_CSSC)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_cvadp_support

Implement instruction to support cache clean by deep persistence (DC CVADP) from ARMv8.5, can be selected for core implemented on any arch version starting ARMv8.2 (FEAT_DPB, FEAT_DPB2).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_data_alignment_flag

Implement non-optimal misalignment flag for PMU/SPE from ARMv8.5

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_debug_rom

If true, a debug ROM will be generated describing the cluster's debug components.

Type

bool

Default value

0x1

has_delayed_brbe_records

If true, a synchronization barrier is required to update the BRB records (FEAT_BRBE).

Type

bool

Default value

0x1

has_delayed_ctireg

Delay the functional effect of CTI register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_dbgreg

Delay the functional effect of external debug register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_mdscr_el1

Delay the functional effect of MDSCR_EL1 register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_oslar_el1

Delay the functional effect of OSLAR_EL1 register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_pmureg

Delay the functional effect of PMU register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_sysreg

Delay the functional effect of system register writes until ISB or implicit barrier.

Type

bool

Default value

0x0

has_delayed_wfe_trap

Implements Configurable Delayed WFE trapping from ARMv8.6 (FEAT_TWED).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_dgh

Implements Data Gathering Hint instruction from ARMv8.6 (FEAT_DGH).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_dot_product

Implement the dot product (UDOT, SDOT) instructions from ARMv8.4 (FEAT_DotProd).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_e0pd

Implement ARMv8.5 feature to prevent unprivileged access to one half of the memory (FEAT_E0PD).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_e2h_rao

Whether the implementation treats HCR_EL2.E2H as Read-As-One (RAO). 0 : FEAT_E2H0 implemented

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.1 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_ebep

Implement Exception-Based Event Profiling from ARMv9.4 (FEAT_EBEP).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_ebf16

Support for Extended BFloat16 Behaviours (FEAT_EBF16)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_ecbhb

Implement Exploitative Control using Branch History information between exception levels (FEAT_ECBHB).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_edacr

Implement EDACR register

Type

bool

Default value

0x1

has_edhsr

Implement external debug halt status register (FEAT_EDHSR). 0: FEAT_EDHSR is not implemented unless architecturally required by another feature, 1: FEAT_EDHSR is implemented, 2: FEAT_EDHSR is implemented (extends EDHSR to include the VNCR, CM, and WnR fields), 0xF: FEAT_EDHSR implementation is dependent on FEAT_SME.

Type

int

Default value

0xf

has_e12

Implements EL2

Type

bool

Default value

0x1

has_e13

Implements EL3

Type

bool

Default value

0x1

has_enhanced_pac

If pointer authentication is enabled then implement enhanced PAC (FEAT_EPAC).

Type

bool

Default value

0x0

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_enhanced_software_step

Implement Enhanced Software Step Extension (FEAT_STEP2).

Possible values of this parameter are: - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_ets

Whether Enhanced Translation Synchronization is supported (FEAT_ETS).

Type

bool

Default value

0x0

has_exception_trapping_form_of_vector_catch

Implement the exception trapping form of vector catch debug event.

Type

bool

Default value

0x1

has_extended_recp_estimate

Implements increased precision of reciprocal instructions (FEAT_RPRES).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_external_rndr

Implement external random number generator module. When enabling this with has_rndr enabled, the external random number generator will be used instead of internal random number generator

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_faminmax

Implement FEAT_FAMINMAX

Type

bool

Default value

0x0

has_far_not_valid

Implements FnV bit in ESR_ELx and xFSR, FAR not valid for synchronous external aborts.

Type

bool

Default value

0x0

has_far_not_valid_dfsc

Implements FnV bit in ESR_ELx, FAR not valid for synchronous external aborts for Data Abort.

Type

bool

Default value

0x0

has_far_not_valid_ifsc

Implements FnV bit in ESR_ELx and xFSR, FAR not valid for synchronous external aborts for Instruction Abort.

Type

bool

Default value

0x0

has_fgt

Implements Fine-grained Virtualization Traps extension from ARMv8.6 (FEAT_FGT).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_fgt2

Implement additional FGT traps introduced in ARMv8.9 (FEAT_FGT2).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_fgwt3

If true then Fine Grained Write EL3 is enabled (FEAT_FGWTE3)

Type

bool

Default value

0x0

has_fixed_function_instr_counter

Implement fixed-function instruction counter (FEAT_PMUv3_ICNTR)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_fp16

Implement the half-precision floating-point data processing instructions from ARMv8.2 (FEAT_FP16).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_fp16_fmlal

Implement the New Floating Point Multiplication Variant (FP16 FMLAL, FMLSL) instructions from ARMv8.4. Only supported if has_fp16=0x1 (FEAT_FHM).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_fpmr

Implement FPMR (FEAT_FPMR)

Type

bool

Default value

0x0

has_frint

Implement floating-point rounding to int instructions (FRINT[32|64][X|Z]) from ARMv8.5 (FEAT_FRINTTS).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_gcs

Implement Guarded Control Stack Extension from ARMv9.4 (FEAT_GCS).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_generic_authentication

Implement ARMv8.3 generic authentication.

Possible values of this parameter are: - 1, feature is implemented if ARMv8.3 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_guest_translation_granule

Implement mechanism for guest translation granule identification from ARMv8.5, ID values determined by stage1 granule configuration parameters (FEAT_GTG).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_haft

Implement Hardware managed Access Flag for Table Descriptors (FEAT_HAFT)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_hardware_accelerator_for_cleaning_dirty_state

Whether hardware accelerator for cleaning Dirty state is supported (FEAT_HACDBS).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_hardware_dirty_state_tracking_structure

Whether hardware Dirty state tracking Structure is supported (FEAT_HDBSS).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_hardware_translation_table_update

Type of hardware translation table supported (when enabled by hardware_translation_table_update_implemented). 0, not implemented. 1, access bit updates implemented. 2, access bit updates and dirty bit mechanism implemented (FEAT_HAFDBS).

Type

int

Default value

0x2

has_hcrx_el2

Implements new HCRX_EL2 id register from Armv8.7 (FEAT_HCX).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_hpmn0

Allow hypervisor to set MDCR_EL2.HPMN to 0 (FEAT_HPMN0)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_id_reg_read

Implement read access to the ID registers (ESR_ELx.EC=0x18) (FEAT_IDST)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_iesb

Implement support for implicit error sync event from ARMv8.2 (FEAT_IESB).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_itd

Implement the optional IT disable feature.

Type

bool

Default value

0x1

has_ite

Implement Instrumentation Trace Extension from ARMv9.4 (FEAT_ITE).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_jscvt

Implement ARMv8.3 javascript Floating-point to Integer conversion instruction (FEAT_JSCVT).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.3 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_large_system_ext

Implement the ARMv8 Large System Extensions (FEAT_LSE).

Type

bool

Default value

0x0

has_large_ttbr_ba_without_lpa

When FEAT_LPA is not implemented, whether TTBR base address supports large values (52 bits) or not (48 bits)

Type

bool

Default value

0x1

has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_ldapur_stlur

Implement support for LDAPR and STLUR instructions with immediate offsets from ARMv8.4 (FEAT_LRCPC2).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_ldm_stm_ordering_control

Implement the SCTLR_ELx.LSMAOE (Load/Store Multiple Atomicity and Ordering Enable) and SCTLR_ELx.nTLSMD (no Trap Load/Store Multiple to Device) controls from ARMv8.2 (FEAT_LSMAOC).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_lrcpc

If true then it support the RCpc feature from ARMv8.3 (FEAT_LRCPC).

Type

bool

Default value

0x0

has_lrcpc3

Implement Release Consistency processor consistent (RCpc) feature from Armv8.9 (FEAT_LRCPC3).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_lut

Implement FEAT_LUT

Type

bool

Default value

0x0

has_mismatch_and_range_breakpoints

Implement Mismatch and Range Breakpoints (FEAT_BWE)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_mismatch_watchpoints

Implement Breakpoints and Watchpoints Enhancements (FEAT_BWE2).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_mops_option

Implement Armv8.8 standard instructions for memory operations (FEAT_MOPS). 0, not implemented (unsupported if Armv8.8 is enabled). 1, implemented using Option A. 2, implemented using Option B

Type

int

Default value

0x0

has_mpam

Implement ARMv8.4 MPAM Registers and associated functionality (FEAT_MPAM).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_mpm

Implement max-power mitigation mechanism (MPMM)

Type

bool

Default value

0x0

has_mt_pmu_disable_feature

Implements Multi-threading PMU disable extension from ARMv8.6 (FEAT_MTPMU). 0: FEAT_MTPMU is disabled, 1: FEAT_MTPMU is enabled if ARMv8.6 is implemented, 2: FEAT_MTPMU is cherry-picked, 0xF: The feature is disabled and is represented by value 0xF in ID_AA64DFR0_EL1.MTPMU.

Type

int

Default value

0x0

has_mte_async_faults

Whether MTE asynchronous faults are supported (FEAT_MTE_ASYNC)

Type

bool

Default value

0x1

has_mte_ctrl_bits_stateful

if memory_tagging_support_level == 1, Whether mte specific control bits in system registers are stateful

Type

bool

Default value

0x0

has_mte_perm

Implement tag access permission (FEAT_MTE_PERM)

Type

bool

Default value

0x0

has_mte_tag_related_fault_high_prio_than_data

For DC GZVA, Whether MMU faults generated by tag access has higher priority than faults due to data access

Type

bool

Default value

0x0

has_nested_virtualization

Implement ARMv8.3 nested virtualization (FEAT_NV).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.3 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_nmi

Implement AARCH64 Non-Maskable Interrupts (FEAT_NMI)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_no_os_double_lock

Do not implement the OS double-lock (FEAT_DoubleLock).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_non_context_synchronizing_exception_controls

Implement cosmetic controls for whether exception entry and exit are context synchronizing events (SCTLR_ELx.{EIS,EOS}) from ARMv8.5 (FEAT_ExS).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_nv1_raz

Whether the implementation treats HCR_EL2.NV1 as Read-As-Zero (RAZ), if has_e2h_rao = 1.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.3 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_nv_frac

Whether the NV_frac behavior is supported.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_par_bit10_razwi

Whether PAR_EL1[10] is RAZ/WI

Type

bool

Default value

0x0

has_partial_delayed_mdscr_el1

has_delayed_oslar_el1 only apply to some bits of MDSCR_EL1 (MDE, KDE, TDCC, SS).

Type

bool

Default value

0x0

has_pc_sample_based_profiling

If true, pc sample-based profiling is enabled (FEAT_PCSRv8, FEAT_PCSRv8p2).

Type

bool

Default value

0x1

has_pc_sample_profiling_enable

Whether PC Sample profiling enable is implemented (FEAT_PCSRv8p9)

Type

bool

Default value

0x0

has_per_cluster_debug_auth_ports

If true then the debug authentication ports i.e. spniden, niden, rpliden, rtpiden, dbgen, spiden are available per cluster

Type

bool

Default value

0x0

has_permission_indirection_s1

Implement the Permission Indirection Extension at stage 1 (FEAT_S1PIE)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_permission_indirection_s2

Implement the Permission Indirection Extension at stage 2 (FEAT_S2PIE)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_permission_overlay_s1

Implement the Permission Overlay Extension at stage 1 (FEAT_S1POE)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_permission_overlay_s2

Implement the Permission Overlay Extension at stage 2 (FEAT_S2POE)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_pmss

Implement PMU Snapshot Extension from Armv8.9 (FEAT_PMUv3_SS).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_pmu

Implement the optional Performance Monitors Extension (FEAT_PMUv3). 0, Not Implemented. 1, Implemented. 2, PMU is IMPLEMENTATION_DEFINED, PMU version would be set to 0xF and would behave as if no PMU is implemented

Type

int

Default value

0x1

has_pmu_edge_detection

Implement PMU Event edge detection (FEAT_PMUv3_EDGE)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_pmu_for_sme_extension

Implement PMUv3 for Scalable Matrix Extension (SME) from ARMv9.5 (FEAT_PMUv3_SME)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_pmu_threshold_linking_control

Implement PMU threshold linking control (FEAT_PMUv3_TH2)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_pointer_authentication

Implement ARMv8.3 pointer authentication (FEAT_PAuth).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.3 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_prediction_invalidation_instructions

Implement execution and data prediction invalidation from ARMv8.5 (FEAT_SPECRES).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_prfm_slc

Implement PRFM with SLC hint (FEAT_PRFMSLC).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_pstate_dit

Implement timing insensitivity of data processing instructions from ARMv8.4 (FEAT_DIT).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_pstate_pan

Implement the PSTATE.PAN (Privileged Access Never) control from ARMv8.1 (FEAT_PAN)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.1 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_pstate_uao

Implement the PSTATE.UAO (User Access Override) control from ARMv8.2 (FEAT_UAO).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_qarma3_pac

Supports QARMA3 pointer authentication algorithm (FEAT_PACQARMA3)

Type

bool

Default value

0x0

has_ras

Implements the ARMv8 RAS Extension. 0 = NO_RAS, 1 = MINIMAL_RAS, 2 = FULL_RAS (FEAT_RAS)

Type

int

Default value

0x0

has_ras_aderr

Implement RAS Asynchronous Device Read Error from Armv8.9 (FEAT_ADERR).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_ras_anerr

Implement RAS Asynchronous Normal Read Error from Armv8.9 (FEAT_ANERR).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_ras_armv84_extension

Implement ARMv8.4 RAS Extension (FEAT_RASv1p1).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_ras_armv89_double_fault

Implement RAS Double Fault Extension from Armv8.9 (FEAT_DoubleFault2).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_ras_armv89_extension

Implement RAS extension from Armv8.9 (FEAT_RASv2).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_ras_critical_error

[DEPRECATED: Set CI field on first register in error_record_feature_register JSON instead]
ARMv8.4 AArch64 RAS Critical Error is implemented or not. 0 - Feature Not Supported, 1 - Feature always enabled, 2 - Feature is controllable.

Type

int

Default value

0x0

has_ras_delegated_serror_exceptions_for_el3

Implement Delegated SError exceptions for EL3 (FEAT_E3DSE).

Possible values of this parameter are: - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_ras_double_fault

Implement ARMv8.4 RAS Double Fault Extension (FEAT_DoubleFault).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_ras_fault_injection

[DEPRECATED: Set INJ field on first register in error_record_feature_register JSON instead]
Implement ARMv8.4 Standard Fault Injection mechanism.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_ras_pfar

Implement RAS Physical Fault Address Registers from Armv8.9 (FEAT_PFAR).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_ras_timestamp

[DEPRECATED: Set TS field on first register in error_record_feature_register JSON instead] ARMv8.4 AArch64 RAS Timestamp register is implemented or not. 0 - No Timestamp is recorded, 1 - Generic Timer timestamp is recorded, 2 - IMP DEF timestamp is recorded.

Type

int

Default value

0x0

has_restriction_on_speculative_data_loaded

Implements the ARMv8.5 security feature (Restrictions on the effects of speculation) (FEAT_CSV3).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_rme

If true, implements full realm management extension (FEAT_RME). Note: This parameter is deprecated and will be removed in future releases, please use rme_support_level parameter.

Type

bool

Default value

0x0

has_rme_gpc2

If true then RME GPC2 extension is enabled (FEAT_RME_GPC2)

Type

bool

Default value

0x0

has_rndr

Implement random number instructions to read from RNDR and RNDRSS random number registers from ARMv8.5 (FEAT_RNG).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_rndr_trap

Implement trapping for RNDR and RNDRSS random number registers from ARMv8.8. (FEAT_RNG_TRAP)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_rounding_doubling_multiply_add_subtract

Implement the rounding doubling multiply add and subtract instructions from ARMv8.1 (FEAT_RDM).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.1 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_rprfm

Support for RPRFM hint instruction (FEAT_RPRFM)

Type

bool

Default value

0x0

has_sctlr2

Implement SCTL2_ELx registers (FEAT_SCTL2).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_sebep

Implement Synchronous-Exception-Based Event Profiling from ARMv9.4 (FEAT_SEBEP).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_secure_el2

Implement support for Secure EL2 (FEAT_SEL2).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_self_hosted_trace_extension

Implement support for the Self-hosted Trace Extensions from ARMv8.4 (FEAT_TRF).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_small_page_table

Implement small page table support which increases the maximum value of TxSZ field from ARMv8.4 (FEAT_TTST). Note: will be unimplemented only if both has_small_page_table=0x0 and has_secure_el2=0x0.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_software_lock

Implement software lock in memory-mapped CTI, PMU, and external debug interfaces

Type

bool

Default value

0x1

has_spe_eft

Implement SPE extended operation type filtering from ARMv9.5 (FEAT_SPE_EFT)

Possible values of this parameter are: - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_spe_fds

Implement SPE filter by data source from ARMv8.9 (FEAT_SPE_FDS)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_spe_for_sme_extension

Implement support of SME to SPE from ARMv9.5 (FEAT_SPE_SME).

Possible values of this parameter are: - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_spe_fpf

Implement SPE operation type extension for ASIMD and FP from ARMv9.5 (FEAT_SPE_FPF)

Possible values of this parameter are: - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_speculation_barrier_inst

Implement speculation barrier instruction (SB) from ARMv8.5 (FEAT_SB).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_speculative_sei

If true, the PE can generate SError interrupt exceptions from speculative reads of memory, including speculative instruction fetches

Type

bool

Default value

0x0

has_stage2_ap_speculative_update

Speculative update of S2 AP bit on S1 TTW. 0 = No Update, 1 = Update, 2 = Update including for AT ops

Type

int

Default value

0x0

has_stage2_fwb

Implement HCR_EL2.FWB, stage 2 control of memory types and cacheability (FEAT_S2FWB)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_stage2_xnx

Implement the extended XN[1:0] stage 2 control from ARMv8.2 (FEAT_XNX).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_stage2_xnx_in_aarch32

Implement the extended XN[1:0] stage 2 control from ARMv8.2 for Aarch32 (FEAT_XNX).

Type

bool

Default value

0x1

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

has_supersections

Whether VMSAv8-32 supersection to support more than 32-bit PA using short descriptor is implemented.

Type

bool

Default value

0x1

has_sve

Whether SVE is implemented (FEAT_SVE). Note: this is required to enable SME (FEAT_SME) with sve.has_sme=1. An SME only implementation can be enabled by setting both as well as sve.sme_only=1

Type

int

Default value

0x0

has_synchronous_load_atomics

Report asynchronous abort due to unsupported load atomics as synchronous (Cacheable)

Type

bool

Default value

0x1

has_synchronous_load_atomics_noncacheable

Report asynchronous abort due to unsupported load atomics as synchronous (Non-Cacheable)

Type

bool

Default value

0x1

has_synchronous_store_atomics

Report asynchronous abort due to unsupported store atomics as synchronous (Cacheable)

Type

bool

Default value

0x0

has_synchronous_store_atomics_noncacheable

Report asynchronous abort due to unsupported store atomics as synchronous (Non-Cacheable)

Type

bool

Default value

0x0

has_sysinstr128

Support for System Instructions that can take 128-bit inputs (FEAT_SYSINSTR128)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_sysreg128

Support for 128-bit System Registers (FEAT_SYSREG128)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_tcr2

Implement TCR2_ELx registers (FEAT_TCR2)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_tidcp1

Implement Armv8.8 ELO use of implementation defined functionality (FEAT_TIDCP1)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_tlb_conflict_abort

Detected inconsistent TLB content generate aborts.

Type

bool

Default value

0x0

has_tlb_pa_caching

Whether intermediate caching of translation table walks might include NonCoherent caches of previous valid walks. 0, NonCoherent caches might be included. 1, No NonCoherent caches included (FEAT_nTLBPA)

Type

bool

Default value

0x0

has_tlbi_range

Implement support for TLB Range Maintenance instructions (TLBI RVAE1, etc) from ARMv8.4 (FEAT_TLBI RANGE).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_tlbi_to_outer_shareable

Implement support for TLB Maintenance instructions that extend to the Outer Shareable domain (TLBI VAE1OS, etc) from ARMv8.4 (FEAT_TLBIOS).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_tlbi_ttl

Implement support for the TTL level hint in by-address TLB Maintenance instructions from ARMv8.4 (FEAT_TTL).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_tlbiw

Implement TLBI instruction for stage2 dirty (FEAT_TLBIW).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_tme

If true, implements TME, the Transactional Memory Extension (FEAT_TME).

Type

bool

Default value

0x0

has_translation_hardening

Implement the Translation Hardening Extension (FEAT_THE)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_trbe

If true, implements the Trace Buffer Extension (FEAT_TRBE).

Type

bool

Default value

0x0

has_trbe_ext

Implements the Trace Buffer external mode extension (FEAT_TRBE_EXT).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv9.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_unaligned_single_copy_atomicity

Implement support for SCTL_R_EL_x.nAA from ARMv8.4, and A64 atomic, exclusive and acquire/release instructions accessing unaligned bytes inside a 16byte window will not generate alignment fault (FEAT_LSE2).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_unsupported_exclusive_fault

Report unsupported exclusive access with Unsupported Exclusive fault status (otherwise use external abort)

Type

bool

Default value

0x1

has_v8_4_debug_extension

Implement ARMv8.4 debug extensions (FEAT_Debugv8p4)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_4_pmu_extension

Implement PMU extension from ARMv8.4 (FEAT_PMUv3p4).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_5_debug_over_power_down

Implement ARMv8.5 Debug over powerdown (FEAT_DoPD).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_v8_5_pmu_extension

Implement PMU extension from ARMv8.5 (FEAT_PMUv3p5).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_5_spe_extension

Implement SPE extension from ARMv8.5 (FEAT_SPEv1p1).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_6_pmu_events

Implements PMU events from ARMv8.6

Possible values of this parameter are: - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_7_fp_enhancements

Implements the Floating Point enhancements from Armv8.7 (introduces FPCR.FIZ/AH/NEP, etc. (FEAT_AFP).)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_7_pmu_events

Implement PMU events from ARMv8.7.

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_7_pmu_extension

Implement PMU extension from ARMv8.7 (FEAT_PMUv3p7).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_7_spe_extension

Implement SPE extension from ARMv8.7 (FEAT_SPEv1p2)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_7_spe_inverted_filtering

Where FEAT_SPEv1p2 is implemented, whether the inverting filtering by events

Type

bool

Default value

0x1

has_v8_7_spe_previous_branch_target

Where FEAT_SPEv1p2 is implemented, whether the optional branch target feature is implemented

Type

bool

Default value

0x1

has_v8_8_debug_extension

Implement ARMv8.8 debug extensions (FEAT_Debugv8p8)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_8_pmu_events

Implement PMU events from ARMv8.8 (FEAT_PMUv3).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_8_pmu_extension

Implement PMU extension from ARMv8.8 (FEAT_PMUv3p8).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_8_spe_extension

Implement SPE extension from ARMv8.8 (FEAT_SPEv1p3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_9_debug_extension

Implement ARMv8.9 debug extensions (FEAT_Debugv8p9)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_9_pc_sample_based_profiling

Implement PC Sample-based Profiling from ARMv8.9 (FEAT_PCSRv8p9)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_9_pmu_events

Implement PMU events from ARMv8.9 (FEAT_PMUv3).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_9_pmu_extension

Implement PMU extension from ARMv8.9 (FEAT_PMUv3p9).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_v8_9_spe_extension

Implement SPE extension from ARMv8.9 (FEAT_SPEv1p4)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_vnocr_el2

Implement support for nested virtualization enhancements from ARMv8.4 (FEAT_NV2).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_wfet_and_wfit

Implements WFE and WFI with Timeout from Armv8.7 (FEAT_WFXT)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_writebuffer

Implement write accesses buffering before L1 cache. May affect ext_abort behaviour.

Type

bool

Default value

0x0

has_xs

Implements Armv8.7 XS, TLBInXS, DSBnXS instruction (FEAT_XS).

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

hcptr_tta_behaviour

Behaviour of HCPTR.TTA when there is no CP14 ETM interface. 0, RAZ/WI. 1, RAO/WI. 2, stateful.

Type

int

Default value

0x2

hcr_el2_miocnce_is_rw

If true, HCR_EL2.MIOCNC is treated as R/W instead of RAZ/WI

Type

bool

Default value

0x0

hcr_swio_res1

Whether HCR.SWIO and/or HCR_EL2.SWIO are RES1.

Type

bool

Default value

0x0

hdbss_error_fault_type

Type of fault reported for HDBSS errors. 0 = precise exception, 1 = fault logged in HDBSSPROD_EL2.FSC (FEAT_HDBSS).

Type

int

Default value

0x0

hpfar_unknown_when_ipa_invalid

If true, HPFAR_EL2 is set to 0 when IPA is not valid for stage 2 faults.

Type

bool

Default value

0x0

hsr_uncond_cc

Condition codes reported in HSR as AL if it passes

Type

bool

Default value

0x0

hvbar_reset_is_rvbar

If true then the reset value of HVBAR is RVBAR, if false the reset value is UNKNOWN.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-log2linelen

If nonzero, Log2 of the instruction cache line length in bytes (valid values in range 4-8). Otherwise the value of cache-log2linelen is used.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-nprefetch

Number of next sequential instruction cache lines to prefetch. This is only used when icache-prefetch_enabled=true.

Type

int

Default value

0x1

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-prefetch_level

0 based cache level at which instructions are pre-fetched. This is only used when icache-prefetch_enabled=true.

Type

int

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or

miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

`icache-read_bus_width_in_bytes`

L1 I-Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

`icache-read_latency`

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

`icache-size`

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

`icache-state_modelled`

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

`icache-ways`

L1 I-Cache number of ways (sets are implicit from size).

Type

int

Default value

0x2

id_spec_fpacc_raz

If true, implementation opts not to disclose the speculative use of pointers processed by a PAC authentication failure by having value 0 for Spec_FPACC bits of ID_AA64MMFR3_EL1 register.

Type

bool

Default value

0x0

ignore_DBGPRCR_CWRR

Ignore writes to the deprecated DBGPRCR.CWRR bit.

Type

bool

Default value

0x0

ignore_access_flag_update_by_at_ops

If true, AT operations do not update access flag

Type

bool

Default value

0x0

ignore_data_abt_on_af_update_by_at_ops

If true, Data abort generated on AF update by AT operations are ignored. This parameter is only valid if ignore_access_flag_update_by_at_ops is false.

Type

bool

Default value

0x1

ignore_large_address_top_bits_in_page_walk

Whether page table bits [15:12] are ignored if PA_SIZE < 52 and output address is configured < 52 with large page

Type

bool

Default value

0x0

ignore_tag_check_dcc_load_store_in_ma_mode_when_tco_is_disabled

Constrained unpredictable behavior for reads/writes to external debug interface DTR regs in memory access mode when PSTATE.TCO is 0. If true, tag check is ignored else, tag check is performed if required.

Type

bool

Default value

0x0

ignore_traps_to_dcc_regs_in_debug

Whether traps get ignored for the following registers in debug state: * AArch64: MDCCSR_EL0, OSDTRENX_EL1, OSDTRTX_EL1, MDCCINT_EL1. * AArch32: DBGDSCRint, DBGDIDR, DBGDSAR, DBGDRAR, DBGDTRRXext, DBGDTRTXext, DBGDCCINT.

Type

bool

Default value

0x0

imp_def_functionality_behaviour

Behaviour of IMPLEMENTATION DEFINED registers and system instructions. 0, UNDEF. 1, RAZ/WI.

Type

int

Default value

0x0

impdef_regs_and_unpred_from_implementation

Configure implementation defined registers and unpredictable behaviour to match the specified implementation. Requires a license for the selected implementation model. User has to provide the default values for the published or configurable parameters through commandline arguments. Use ARM_Cortex-A<num> or ARM_<codename> for licensed pre-release cores.

Type

string

Default value**impdef_sysreg_json**

Configure mask/reset bitmasks for impdef. registers in a JSON format which is (where 'bitwise' indicates 'reset'/'mask' to flip existing bits): [{ "name": "IMP_SYSREG0_EL1", "reset": 0, "mask": 0, "encoding": 0, "bitwise": true}]

Type

string

Default value

[]

independent_cache_control_traps

Implement Independent Cache Control traps from ARMv8.5. 0, No support. 1, Supported but not for tlb maintenance instructions. 2, Full support. (FEAT_EVT)

Type

int

Default value

0x0

insert_iesb_before_exception

If true then inserts an IESB before taking with Exception otherwise has no effect and IESB is taken after PState is changed due to the Exception.

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

internal_vgic

Instantiate VGIC peripheral in this processor

Type

bool

Default value

0x0

is_debug_state_pmu_snapshot_allowed

If true, PMU snapshot is allowed in debug state.

Type

bool

Default value

0x1

is_first_pcsr_sample_ignored

If true, First read of PMPCSR register after reset returns 0xFFFFFFFF

Type

bool

Default value

0x0

is_mt_res0

If ARMv8.6 is not implemented, and PMUV3 is implemented, this parameter controls whether PMEVTYPER<n>.MT bit is RES0 or RW. For other implementations, this parameter has no effect

Type

bool

Default value

0x0

is_serror_edge_triggered

If true, SError is edge-triggered. Otherwise, its level-triggered.

Type

bool

Default value

0x1

is_tagged_nsh_treated_as_tagged

Whether a tagged NonShared memory attribute is treated as tagged or not

Type

bool

Default value

0x1

is_uniprocessor

Value for the U bit in MPIDR. true disables L1 cache coherency protocols

Type

bool

Default value

0x0

isb_is_branch

If true, ISB is traced as an immediate branch. This allows to count ISB as a branch in debug extensions (e.g. PMU)

Type

bool

Default value

0x0

ish_is_osh

Whether Innershareable is same as OuterShareable

Type

bool

Default value

0x0

itd_conditional_instructions_are_32bit

When SCTLx_ELx.ITD=1, an IT instruction plus a T16 instruction are considered a single 32bit conditional instruction.

Type

bool

Default value

0x0

jidr_is_undef_at_el0

If true, JIDR register access is UNDEF at EL0

Type

bool

Default value

0x0

jmcr_is_undef_at_el0

If true, JMCR register access is UNDEF at EL0

Type

bool

Default value

0x0

joscr_is_undef_at_el0

If true, JOSCR register access is UNDEF at EL0

Type

bool

Default value

0x0

l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-mpamf.arch_major_ver

L3 Cache MPAMF_AIDR architecture major version

Type

int

Default value

0x0

l3cache-mpamf.arch_minor_ver

L3 Cache MPAMF_AIDR architecture minor version

Type

int

Default value

0x0

l3cache-mpamf.bwa_width_ns

L3 Cache width of MPAM bandwidth allocation fields for non-secure accesses.

Type

int

Default value

0x10

l3cache-mpamf.bwa_width_s

L3 Cache width of MPAM bandwidth allocation fields for secure accesses.

Type

int

Default value

0x10

l3cache-mpamf.cmax_width_ns

L3 Cache number of fractional bits in MPAM cache capacity partition control for non-secure accesses. Only the register interface is implemented - the control is NOT FUNCTIONAL.

Type

int

Default value

0x0

l3cache-mpamf.cmax_width_s

L3 Cache number of fractional bits in MPAM cache capacity partition control for secure accesses. Only the register interface is implemented - the control is NOT FUNCTIONAL.

Type

int

Default value

0x0

l3cache-mpamf.cpbm_width_ns

L3 Cache width of MPAM cache portion bitmap for non-secure accesses. If 0, the feature is not implemented, and all ways are available.

Type

int

Default value

0x0

l3cache-mpamf.cpbm_width_rl

L3 Cache width of MPAM cache portion bitmap for realm accesses. If 0, the feature is not implemented, and all ways are available.

Type

int

Default value

0x0

l3cache-mpamf.cpbm_width_rt

L3 Cache width of MPAM cache portion bitmap for root accesses. If 0, the feature is not implemented, and all ways are available.

Type

int

Default value

0x0

l3cache-mpamf.cpbm_width_s

L3 Cache width of MPAM cache portion bitmap for secure accesses. If 0, the feature is not implemented, and all ways are available.

Type

int

Default value

0x0

l3cache-mpamf.csu_num_mon_ns

L3 Cache number of MPAM cache storage usage monitors for non-secure accesses.

Type

int

Default value

0x0

l3cache-mpamf.csu_num_mon_rl

L3 Cache number of MPAM cache storage usage monitors for realm accesses.

Type

int

Default value

0x0

l3cache-mpamf.csu_num_mon_rt

L3 Cache number of MPAM cache storage usage monitors for root accesses.

Type

int

Default value

0x0

l3cache-mpamf.csu_num_mon_s

L3 Cache number of MPAM cache storage usage monitors for secure accesses.

Type

int

Default value

0x0

l3cache-mpamf.has_esr

L3 Cache's MPAMF_ESR, MPAMF_ECR, and MPAM error handling implemented.

Type

bool

Default value

0x0

l3cache-mpamf.has_extd_esr

L3 Cache's MPAMF_ESR is 64-bits.

Type

bool

Default value

0x0

l3cache-mpamf.has_impl_idr

L3 Cache's MPAMF_IMPL_IDR is present.

Type

bool

Default value

0x0

l3cache-mpamf.has_mbwu_long_counter

L3 Cache has long MBWU counter and capture registers.

Type

bool

Default value

0x0

l3cache-mpamf.has_mpamfidr_ext

MPAMF_IDR.EXT support

Type

bool

Default value

0x0

l3cache-mpamf.has_partid_nrw

Narrowing part ID register is present. This is global rather than per-instance.

Type

bool

Default value

0x0

l3cache-mpamf.has_priority_partitioning

The selected resource has priority partitioning described in MPAMF_PRI_IDR.

Type

bool

Default value

0x0

l3cache-mpamf.has_prod_id

L3 Cache MPAMF_IIDR product ID supported.

Type

int

Default value

0x0

l3cache-mpamf.has_prod_rev

L3 Cache MPAMF_IIDR product REVISION supported.

Type

int

Default value

0x0

l3cache-mpamf.has_prod_var

L3 Cache MPAMF_IIDR product VARIANT supported.

Type

int

Default value

0x0

l3cache-mpamf.has_prop_ns

Enable memory bandwidth proportional stride control for non-secure accesses. Only the register interface is implemented - the control is NOT FUNCTIONAL.

Type

bool

Default value

0x0

l3cache-mpamf.has_prop_s

Enable memory bandwidth proportional stride control for secure accesses. Only the register interface is implemented - the control is NOT FUNCTIONAL.

Type

bool

Default value

0x0

l3cache-mpamf.has_ris

L3 Cache has resource instance selection support.

Type

bool

Default value

0x0

l3cache-mpamf.max_partid_ns

L3 Cache Maximum value of non-secure PARTID supported.

Type

int

Default value

0xffff

l3cache-mpamf.max_partid_rl

L3 Cache Maximum value of realm PARTID supported for RME implementations.

Type

int

Default value

0xffff

l3cache-mpamf.max_partid_rt

L3 Cache Maximum value of root PARTID supported for RME implementations.

Type

int

Default value

0xffff

l3cache-mpamf.max_partid_s

L3 Cache Maximum value of secure PARTID supported.

Type

int

Default value

0xffff

l3cache-mpamf.max_pmg_ns

L3 Cache Maximum value of non-secure PMG supported.

Type

int

Default value

0xff

l3cache-mpamf.max_pmg_rl

L3 Cache Maximum value of realm PMG supported for RME implementations.

Type

int

Default value

0xff

l3cache-mpamf.max_pmg_rt

L3 Cache Maximum value of root PMG supported for RME implementations.

Type

int

Default value

0xff

l3cache-mpamf.max_pmg_s

L3 Cache Maximum value of secure PMG supported.

Type

int

Default value

0xff

l3cache-mpamf.mbwu_long_counter_width

L3 Cache long MBWU counter width. 0: 63 bits, 1: 44 bits.

Type

int

Default value

0x0

l3cache-mpamf.no_impl_msmon

L3 Cache's MPAMF_IMPL_IDR does not describe resource monitors.

Type

bool

Default value

0x0

l3cache-mpamf.no_impl_part

L3 Cache's MPAMF_IMPL_IDR does not describe resource partitioning controls.

Type

bool

Default value

0x0

l3cache-mpamf.ris_max

L3 Cache's largest resource instance selector value defined.

Type

int

Default value

0x0

l3cache-mpamf_base

L3 Cache memory mapped MPAM registers base address

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_bus_width_in_bytes

L3 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x0

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_bus_width_in_bytes

L3 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x8

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

legacy_combining_exc_catch_trace

Whether exception catch is traced as part of exception entry/exit in same cycle

Type

bool

Default value

0x1

log2_trace_buffer_alignment

Log2 of trace buffer alignment constraint for output buffer (0->1B ... 11->2Kib)

Type

int

Default value

0x0

ls64_ignore_s1_unpred_memattr_transformation

If true, stage 1 unpredictable memory attribute transformations are ignored for FEAT_LS64 single-copy atomic 64-byte load/store instructions' (FEAT_LS64, FEAT_LS64_V, FEAT_LS64_ACCDATA).

Type

bool

Default value

0x0

ls64_memtype_check_use_combined_memattr

If true, FEAT_LS64 single-copy atomic 64-byte load/store instructions' memory attributes check is done on the combined memory attributes at the end of all enabled translation stages.

Type

bool

Default value

0x0

max_32bit_el

Maximum exception level supporting AArch32 modes. -1: No Support for A32 at any EL, x:
[0:3] - All the levels below supplied ELx supports A32

Type

int

Default value

0x3

mdrar_el1_res0

MDRAR_EL1 is RES0.

Type

bool

Default value

0x0

mdselr_le_16_bps_wps_behaviour

Behaviour of MDSELR_EL1 and related traps/enables if fewer than 16 watchpoints and fewer than 16 breakpoints are implemented: 0 - MDSELR_EL1 is stateful 1 - MDSELR_EL1, EBWE, FGTS are RAZ/WI, traps and enables do not apply

Type

int

Default value

0x0

mec_support_level

0 -> Memory Encryption Contexts not implemented, 1 -> LEGACY_TZ_EN mode i.e. MEC register fields are stateful but only supports secure/non-secure states, 2 -> Memory Encryption Contexts fully implemented (FEAT_MEC).

Type

int

Default value

0x0

memory.acp.AxCACHE_mask

Used with memory.acp.AxCACHE_pattern to define which memory types the ACP port accepts. All transactions which do not satisfy (AxCACHE & mask) == pattern will abort.

Type

int

Default value

0x0

memory.acp.AxCACHE_pattern

Used with memory.acp.AxCACHE_mask to define which memory types the ACP port accepts. All transactions which do not satisfy (AxCACHE & mask) == pattern will abort.

Type

int

Default value

0x0

memory.l2_cache.is_inner_cacheable

L2 cache obeys inner cacheable attributes (rather than outer cacheable attributes)

Type

bool

Default value

0x1

memory.l2_cache.is_inner_shareable

L2 cache obeys inner shareable attributes (rather than outer shareable attributes)

Type

bool

Default value

0x1

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented. 1, instructions and registers only are implemented (FEAT_MTE). 2, implemented (FEAT_MTE2). 3, implemented with asymmetric handling of exceptions (FEAT_MTE3). 4, implemented (FEAT_MTE4).

Type

int

Default value

0x0

mixed_endian

Implement support for mixed endianness. 0, not supported. 1, supported at all exception levels. 2, supported at ELO only.

Type

int

Default value

0x1

mops_cpy_block_size

Block size used for memcpy memory accesses

Type

int

Default value

0x40

mops_cpy_default_dir

Default direction for non-overlapping memcpy operations: 0, forwards. 1, backwards

Type

int

Default value

0x0

mops_cpy_handle_async_exceptions

Handle any pending async exceptions after copying a block of data, instead of waiting until instruction end.

Type

bool

Default value

0x0

mops_cpy_post_size

Percentage of data copied in memcpy 'E' instructions

Type

int

Default value

0xa

mops_cpy_pre_size

Percentage of data copied in memcpy 'P' instructions

Type

int

Default value

0xa

mops_cpy_pre_size_threshold

Size threshold in Bytes for CPY*P* instructions

Type

int

Default value

0x0

mops_cpy_single_access

Execute memcpy as a single read and single write access

Type

bool

Default value

0x0

mops_cpy_write_abort_before_read

Report the data aborts and watchpoint of the write accesses, before those of the read accesses

Type

bool

Default value

0x0

mops_cpy_zero_size_can_fault

Fault because of mismatched implementation option when the operation is of size 0.

Type

bool

Default value

0x1

mops_exec_order_can_fault

Enable exception on the Main/Epilogue instruction when executed after a mismatched Prologue/Main in a CPY/SET sequence, or after another random instruction

Type

bool

Default value

0x0

mops_inst_cpy_zero_size_can_fault

Fault because of mismatched implementation option when inst_cpy_size is 0

Type

bool

Default value

0x1

mops_inst_set_zero_size_can_fault

Fault because of mismatched implementation option when inst_set_size is 0

Type

bool

Default value

0x1

mops_mismatched_page_crossing_access_unpred

Constrained unpredictable behaviour for FEAT_MOPS instructions when crossing page boundary with different memory types, 0 : Memory block access uses the attributes of it's own address block 1: Alignment Fault

Type

int

Default value

0x0

mops_mmu_abort_far_aligned

If true, in case of an MMU abort on a MOPS instruction, report FAR aligned to current translation granule.

Type

bool

Default value

0x0

mops_set_block_size

Block size used for memset memory accesses

Type

int

Default value

0x40

mops_set_handle_async_exceptions

Handle any pending async exceptions after setting a block of data, instead of waiting until instruction end.

Type

bool

Default value

0x0

mops_set_post_size

Percentage of data copied in memset 'E' instructions

Type

int

Default value

0xa

mops_set_pre_size

Percentage of data copied in memset 'P' instructions

Type

int

Default value

0xa

mops_set_single_access

Execute memset as a single read and single write access

Type

bool

Default value

0x0

mops_set_zero_size_can_fault

Fault because of mismatched implementation option when the operation is of size 0.

Type

bool

Default value

0x1

mops_setg_unaligned_does_mismatch_fault

If true, in case of unaligned SETGM / SETGE, raise a mismatched memset exception because of impdef reasons, instead of alignment fault

Type

bool

Default value

0x0

mops_wp_far_behaviour

Set option for address stored in FAR/EDWARD after watchpoints hit by MOPS instructions

- 0 - FAR recorded matches lowest watchpointed address accessed by the instruction
- 1 - FAR recorded matches lowest address accessed by the instruction within same translation granule as watchpointed address
- 2 - FAR recorded matches highest watchpointed address accessed by the instruction that triggered the watchpoint

Type

int

Default value

0x0

mpam_force_ns_rao

Whether MPAM3_EL3.FORCE_NS bit is RAO/WI.

Type

bool

Default value

0x0

mpam_frac

MPAM fractional revision number in ID_AA64PFR1_EL1.MPAM_frac field. Combines with has_mpam to give the mpam version has_mpam = false, mpam_frac = 0 -> Not implemented has_mpam = false, mpam_frac = 1 -> FEAT_MPAMv0p1 has_mpam = true, mpam_frac = 0 -> FEAT_MPAMv1p0 has_mpam = true, mpam_frac = 1 -> FEAT_MPAMv1p1

Type

int

Default value

0x0

mpam_has_altsp

MPAM Whether MPAMIDR_EL1.HAS_ALTSP bit is set or clear

Type

bool

Default value

0x0

mpam_has_hcr

MPAM Whether MPAMIDR_EL1 HAS_HCR bit is set or clear

Type

bool

Default value

0x0

mpam_max_partid

MPAM Maximum PARTID Supported

Type

int

Default value

0xffff

mpam_max_pmg

MPAM Maximum PMG Supported

Type

int

Default value

0xff

mpam_max_vpmr

MPAM Maximum VPMR Supported

Type

int

Default value

0x0

mpamidr_has_force_ns

Whether MPAMIDR_EL1.HAS_FORCE_NS bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

mpamidr_has_sdeflt

Whether MPAMIDR_EL1.HAS_SDEFLT bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

mpamidr_has_tidr

Whether MPAMIDR_EL1.HAS_TIDR bit is set or clear

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

mpidr_layout

Layout of MPIDR. 0 AFF0 is CPUID, 1 AFF1 is CPUID

Type

int

Default value

0x0

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

mpmm_config

MPMMTUNE register value. The JSON schema is : { "DT_THR":0, "MPMM_G2_TP":2, "MPMM_G2_ATHR":100, "MPMM_G1_TP":2, "MPMM_G1_ATHR":75, "MPMM_GO_TP":2, "MPMM_GO_ATHR":50 } . The value given for threshold value is just an indication, not specific to any core. This parameter is used only when has_mpmm is set.

Type

string

Default value**mte_report_which_failed_address**

Set to "first", "last" or "random". Applicable only for MTE synchronous check. If "first" then report first failed address. If "last" then report the first address of last failed MTE granule or first address of transfer if it doesn't cross a granule boundary. If "random" then report the first address of a random failed MTE granule.

Type

string

Default value

first

mte_tminline

Value of CTR_ELO.TminLine for reading purpose only. A value configured using this does not indicate the presence of separate tag cache. 0, TminLine evaluated from smallest data cache line.

Type

int

Default value

0x0

mvbar_reset_is_rvbar

If true then the reset value of MVBAR is RVBAR, if false the reset value is UNKNOWN.

Type

bool

Default value

0x1

non_secure_vgic_alias_when_ns_only

If ! has_el3 and only non-secure side exists, then the normal position of the VGIC is a secure alias. If this parameter is non-zero then in addition a non-secure alias of the VGIC will be placed at this position (aligned to 32 KB).

Type

int

Default value

0x0

num_loregion_descriptors

Number of Limited Ordering Region descriptors implemented (if ARM v8.1 extensions are implemented) (FEAT_LOR).

Type

int

Default value

0x0

num_loregions

Number of Limited Ordering Regions implemented excluding background region (if ARM v8.1 extensions are implemented) (FEAT_LOR).

Type

int

Default value

0x0

number_of_abl_breakpoints

if FEAT_ABLE is implemented, Number of address matching breakpoints that support address linking.

Type

int

Default value

0x0

number_of_error_records

Cores Number of Error records supported for RAS

Type

int

Default value

0x0

optimal_alignment_size

Alignment boundary which does not incur additional performance penalty from ARMv8.5.

- 1, architectural misalignment is used to set PMU event LDST_ALIGN_LAT and SPE event E[11]
- 2, access crossing 4 byte boundary is used to set PMU event LDST_ALIGN_LAT and SPE event E[11]
- 3, access crossing 8 byte boundary is used to set PMU event LDST_ALIGN_LAT and SPE event E[11]
- ... - 12, access crossing 4 Kbyte boundary is used to set PMU event LDST_ALIGN_LAT and SPE event E[11]

Type

int

Default value

0x1

output_attributes

User-defined transform to be applied to bus attributes like MasterID, ExtendedID or UserFlags. Currently, only works for MPAM Attributes encoding into bus attributes.

Type

string

Default value

ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID,
ExtendedID[38]=MPAM_NS

pacm_support_level

Implements PSTATE.PACM from ARMv9.5. 0: Not supported, 1: Trivial implementation when FEAT_PAuth_LR and FEAT_PACIMP are supported, 2: Full implementation when FEAT_PAuth_LR is supported.

Type

int

Default value

0x2

page_based_hardware_attributes

Implement the page based hardware attributes from ARMv8.2. This parameter indicates which page table bits are available for hardware, where bits[3:0] correspond to PTE[62:59] and to TCR_ELx.HWUnyy (FEAT_HPDS2).

Type

int

Default value

0x0

par_ns_set_unknown_bit

Whether NS bit of PAR is set/clear when executing AT to perform non-secure regime translation. When true, NS is set to 1 else 0.

Type

bool

Default value

0x1

pfar_is_valid

IMPLEMENTATION DEFINED choice to configure ESR_ELx.PFV: whether PFAR_ELx is valid or UNKNOWN when ESR_ELx.PFV is not forced to be 0.

Type

bool

Default value

0x1

pfrr1_csv2_frac

Fractional revision number ID_AA64PFR1_EL1.CSV2_frac when ID_AA64PFR0_EL1.CSV==1 for CSV2 extension (FEAT_CSV2_1p1, FEAT_CSV2_1p2)

Type

int

Default value

0x0

pmb_idr_external_abort

Describes how the PE manages External aborts on writes made by the Statistical Profiling Extension to the Profiling Buffer. 0, External abort is reported to SPE, From Armv8.8 and Armv9.3, the value 0 is not permitted. 1, External abort is ignored. 2, The External abort generates an SError and the error is not reported to SPE.

Type

int

Default value

0x0

pmb_idr_flag_updates

Defines whether the address translation performed by the Profiling Buffer manages the Access Flag and dirty state

Type

bool

Default value

0x1

pmcr_disable_events_export

If true, export for PMU events is disabled. This configures PMCFGR.EX field.

Type

bool

Default value

0x1

pmmir_ell_bus_slots

Largest value by which BUS_ACCESS can increment over BUS_CYCLES cycles. From v8.7 PMU extension

Type

int

Default value

0x0

pmmir_ell_bus_width

Width, in bytes, of accesses counted by BUS_ACCESS. From v8.7 PMU extension

Type

int

Default value

0x0

pms_idr_max_size

Defines largest size for a single SPE record (rounded up to a power of 2)

Type

int

Default value

0x6

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x8

pmu_async_exception_delay

Configure PMU asynchronous exception delay in CPU cycles (FEAT_SEBEP)

Type

int

Default value

0x0

pmu_cycle_counter_counts_actual_cycles

If true and Timing annotation is enabled, PMU cycle counter counts actual cycles, otherwise counts instructions executed

Type

bool

Default value

0x0

pmu_has_chain_event

PMU (if present) implements event number 0x1e, CHAIN.

Type

bool

Default value

0x1

pmu_precise_events

"Configure v9.4 Precise PMU events. {"pmu_events":["SW_INCR", "PC_WRITE_RETIRED", "BR_RETIRED", "BR_IND_RETIRED", "BR_RETURN_RETIRED", "BR_RETURN_ANY_RETIRED", "BR_IND_TAKEN_RETIRED", "LD_RETIRED", "ST_RETIRED", "UNALIGNED_LD_ST", "INST_RETIRED", "EXCEP_TAKEN", "EXCEP_RETURN", "CHAIN"]}"

Type

string

Default value**pmu_threshold_bit_width**

Implement FEAT_PMUV3_TH and if so the width of PMEVTYPER<n>_ELO.TH in bits. 0, not implemented. 1-12 number of bits in PMEVTYPER<n>_ELO.TH.

Type

int

Default value

0x0

pseudo_fault_generation_feature_register

ARMv8.4 Standard Pseudo-fault generation feature register values. JSON schema for the parameter value is: [{"OF":false, "UC":false, "UEU":false, "UER":false, "UEO":false, "DE":false, "CE":0x0, "CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":false, "NA":false}, other_pseudo-fault_generating_features_register_values]. Where OF, UC, UEU, UER, UEO, DE, CI, ER, PN, AV, MV, SYN, and R have valid false(NOT_SUPPORTED) and true(FEATURE_CONTROLLABLE), where CE can have 0(NOT_SUPPORTED), 1(NONSPECIFIC_CE_SUPPORTED) and 3(TRANSIENT_OR_PERSISTENT_CE_SUPPORTED) and NA can have false(component fakes detection on next access) or true(component fakes detection spontaneously). Effective only when ERXFR's INJ field allows it or has_ras_fault_injection is true.

Type

string

Default value**pstate_ssbs_reset**

Reset value of pstate.ssbs

Type

bool

Default value

0x0

pstate_ssbs_type

Implement speculative store bypass safe feature from ARMv8.5. 0, Not supported. 1, Supported without MSR/MRS access to SSBS (FEAT_SSBS). 2, fully supported (FEAT_SSBS2).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

randomize_unknowns_at_reset

Will fill in unknown bits in registers at reset with random value using register_reset_data as seed, it overrides scramble_unknowns_at_reset

Type

bool

Default value

0x0

ras_aderr_anerr_controls_are_same

If true and FEAT_ADERR and FEAT_ANERR is implemented then ADERR and ANERR controls should always be set to the same value (FEAT_ADERR) (FEAT_ANERR)

Type

bool

Default value

0x0

ras_err_registers_undef_if_no_error_records

If true, all RAS error record registers, along with ERRSELR_EL1, will be undefined if ERRIDR_EL1 indicates that zero error records are implemented.

Type

bool

Default value

0x0

ras_errselr_undef_if_no_error_records

If true, ERRSELR_EL1 will be undefined if ERRIDR_EL1 indicates that zero error records are implemented.

Type

bool

Default value

0x0

ras_extra_configurations

Miscellaneous configurations for error records. An array of JSON objects. Note for ERXCTLR_EL1 register it only allows to define the mask value for the IMPDEF fields, ie bits [63:32] and bit 1, but its reset value applies on all fields. Note for ERXMISCn masks - these are 64 bit masks covering the 64 bit registers ERXMISCn_EL1. E.g. [{"Index": 0, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXMISC1_mask": 0x0, "ERXMISC1_reset": 0x0, "ERXMISC2_mask": 0x0, "ERXMISC2_reset": 0x0, "ERXMISC3_mask": 0x0, "ERXMISC3_reset": 0x0, "ERXCTLR_EL1_mask": 0x0, "ERXCTLR_EL1_reset": 0x0}, {"Index": 1, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXSTATUS_IERR_mask": 0x300}]

Type

string

Default value

ras_frac

0, No additional feature implemented. 1, Additional ERXMISC*, ERXPFG* registers and FaultInjection trap from RAS v1.1. implemented

Type

int

Default value

0x0

ras_log2_fault_granule_size

Log2 of the RAS fault granule size in KB.

Type

int

Default value

0x2

ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0x18

ras_report_aligned_pa_in_pfar

If true, the PFAR_ELx register reports the PA aligned to the RAS fault granule size on a sync external abort or SError exception.

Type

bool

Default value

0x0

register_reset_data

Data used to fill register bits when they become UNKNOWN at reset.

Type

int

Default value

0x0

register_reset_data_hi

Data used to fill the upper-half of 128-bit registers when the bits become UNKNOWN at reset.

Type

int

Default value

0x0

report_iside_cmo_ifsr

fault info for an iside cache maintenance operation is reported in the IFSR

Type

bool

Default value

0x1

report_second_access_align_fault_non_atomic_pair_access

If true, the non-atomic load/store pair accesses report the 2nd register as faulting address for an alignment fault. This is IMP-DEF behavior as defined in FEAT_LRCPC3.

Type

bool

Default value

0x0

report_second_access_mmu_fault_non_atomic_pair_access

If true, the non-atomic load/store pair accesses report the 2nd register as faulting address for an MMU fault. This is IMP-DEF behavior as defined in FEAT_LRCPC3.

Type

bool

Default value

0x0

reported_fp_revision

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored. Updates the FPSID.revision value

Type

int

Default value

0xffffffffffffffff

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reserved_HMC_SSC_PAC_treated_disabled

When DBG[B|W]CR.{HMC,SSC,PAC} bits configuration is reserved, this parameter controls whether breakpoints/watchpoints are treated as Disabled or not

Type

bool

Default value

0x0

restore_fpsr_on_trapped_fp_exception

If true, FPSR is restored to the value of the FPSR immediately before the instruction that generated the trapped floating-point exception.

Type

bool

Default value

0x0

restriction_on_speculative_execution

Implements the ARMv8.5 security feature (Restrictions on the effects of speculation), ID_AA64PFR0_EL1.CSV2: 0: No disclosure whether branch targets trained in one context can affect speculative execution in a different context, 1: Branch targets trained in one context cannot affect speculative execution in a different hardware described context (SCXTNUM_ELx not supported), 2: Branch targets trained in one context cannot affect speculative execution in a different hardware described context (SCXTNUM_ELx supported) (FEAT_CSV2, FEAT_CSV2_2), 3: FEAT_CSV2_3 is supported

Type

int

Default value

0x0

restriction_on_speculative_execution_aarch32

Implements the ARMv8.5 security feature (Restrictions on the effects of speculation), ID_PFR0.CSV2: 0: No disclosure whether branch targets trained in one context can affect speculative execution in a different context, 1: Branch targets trained in one context cannot affect speculative execution in a different hardware described context, 2: Branch targets trained in one context cannot affect speculative execution in a different hardware described context or at a different address in the same hardware described context (FEAT_CSV2, FEAT_CSV2_2).

Type

int

Default value

0x0

revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

rgsr_res0_stateful

Whether RGSr read of RES0 bits return last written value

Type

bool

Default value

0x0

rme_default_mecid_nonsecure

Default MECID value for NON-SECURE PAS

Type

int

Default value

0x0

rme_default_mecid_realm

Default MECID value for REALM PAS

Type

int

Default value

0x0

rme_default_mecid_root

Default MECID value for ROOT PAS

Type

int

Default value

0x0

rme_default_mecid_secure

Default MECID value for SECURE PAS

Type

int

Default value

0x0

rme_full_is_tagged_nsh_treated_as_tagged

Whether a tagged NonShared memory attribute is treated as tagged or not. Does nothing if effective RME support is not full

Type

bool

Default value

0x0

rme_level0_gpt_size

The range of address space protected by each entry in the level 0 GPT (0->1GB 1->16GB, 2->64GB, 3->512GB)

Type

int

Default value

0x0

rme_mecid_width

Width of MECID in bits

Type

int

Default value

0x1

rme_support_level

0 -> Realm management extension not implemented, 1 -> LEGACY_TZ_EN mode i.e. RME register fields are stateful but only supports secure/non-secure states, 2 -> Realm management extension fully implemented (FEAT_RME).

Type

int

Default value

0x0

rmr_always_implemented

Always implement RMR_ELx, RMR, or HRMR at the highest implemented exception level, even if that exception level cannot use both AArch32 and AArch64.

Type

bool

Default value

0x0

rndr_rndrrs_seed

Initial seed for random engine used in RNDR register

Type

int

Default value

0x0

s1_align_memtype_fault_prio_more_than_s2_perm_fault_on_s1_walk

If true, s1 alignment fault has priority over s2 permission faults

Type

bool

Default value

0x1

scheduler_mode

Control the interleaving of instructions in this processor. 0, default long quantum. 1, low latency mode, short quantum and signal checking. 2, lock-breaking mode, long quantum with additional context switches near load-exclusive instructions. WARNING: This parameter is intended for validation purposes and may result in unwanted behaviour if altered!

Type

int

Default value

0x0

scr_nET_writeable

Whether SCR.nET is writeable. Writing to it is purely cosmetic (nET behavior not implemented)

Type

bool

Default value

0x0

scramble_unknowns_at_reset

Will fill in unknown bits in registers at reset with register_reset_data

Type

bool

Default value

0x1

serror_clear_delay

Delay for clearing of SError if SError is level-triggered, in cpu cycles.

Type

int

Default value

0x0

set_mops_option

Set option for Armv8.8 SET(FEAT_MOPS). 0, use default(i.e. use value configured through has_mops_option). 1, implemented using Option A. 2, implemented using Option B

Type

int

Default value

0x0

set_rasv10_for_armv84_and_higher

ARMv8.4 mandates RAS System Architecture v1.1, but when there are no error records and FEAT_DoubleFault is not implemented then there is no functional difference between the RAS System Architecture v1.0 (that is, the RAS extension as in pre-ARMv8.4 implementations) and the RAS System Architecture v1.1 (also known as FEAT_RASv1p1). This flag if true will set the RAS ID to declare RAS v1.0 rather than RAS v1.1 for ARMv8.4 and higher implementations. If this is set and the core does not conform to the restrictions then this parameter is ignored.

Type

bool

Default value

0x0

setg_mops_option

Set option for Armv8.8 SETG(FEAT_MOPS). 0, use default(i.e. use value configured through has_mops_option). 1, implemented using Option A. 2, implemented using Option B

Type

int

Default value

0x0

skip_trace_on_write_to_oseccr_el1_when_oslock_is_unlocked

If OSLSR_EL1.OSLK == 0, then OSECCR_EL1 returns an unknown value on reads and ignores writes. When true, also skips the traces on writes to OSECCR_EL1 when OSLSR_EL1.OSLK == 0.

Type

bool

Default value

0x0

spe_counter_size

Size of counter packet payload in Statistical Profiling Extension - 1, Counter packet payloads are 12-bit saturating counters - 2, Counter packet payloads are 16-bit saturating counters

Type

int

Default value

0x1

spmu_support_level

Implement System PMU: 0: Not supported, 1: v8.9 System PMU Extension is implemented (FEAT_SPMU), 2: v9.5 System PMU2 Extension is implemented (FEAT_SPMU2)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.9 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

spsr_el3_is_mapped_to_spsr_mon

Whether SPSR_EL3 is mapped to AArch32 register SPSR_mon

Type

bool

Default value

0x1

spsr_m4_res0

Whether SPSR_ELx.M[4] bit should be RES0 for AARCH64 only implementations

Type

bool

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

stage1_tlb_size

Number of stage1 only tlb entries.

Type

int

Default value

0x0

stage1_walkcache_size

Number of stage1 only walk cache entries.

Type

int

Default value

0x0

stage2_tlb_size

Number of stage2 only tlb entries.

Type

int

Default value

0x0

stage2_walkcache_size

Number of stage2 only walk cache entries.

Type

int

Default value

0x0

statistical_profiling_buffer_alignment

Statistical profiling alignment constraint for sample buffer

Type

int

Default value

0x1

statistical_profiling_random_interval_is_separate

Statistical profiling random interval gets added to the main timer interval(false) or (true) runs as separate timer

Type

bool

Default value

0x0

statistical_profiling_recommended_min_sampling

Statistical profiling recommended minimum sampling interval

Type

int

Default value

0x100

stex_fail_suppress_sync_data_aborts

If true, synchronous data aborts are not reported if store exclusive fails

Type

bool

Default value

0x0

store_excl_fail_tag_check_action

Behavior of tag check by core when a store exclusive fails. 0, Tag check ignored, 1, Tag check done if exclusive fails by global monitor.

Type

int

Default value

0x0

strex_fail_can_hit_watchpoint

If true, a strex fail can hit watchpoint

Type

bool

Default value

0x0

stzgm_reports_fault_address_from_reg_arg

Which faulting address should be reported in FAR_ELx on a failed STZGM: 0: the lowest aligned addr to DCZID-log2-block-size, 1: the addr held in the register argument, 2: if it is a tag-check fault, the addr aligned to DCZID-log2-block-size, otherwise the addr held in the register argument.

Type

int

Default value

0x0

supports_multi_threading

Sets the MPIDR.MT bit. Setting this to true hints the the cluster is multi-threading compatible

Type

bool

Default value

0x0

sve.clear_constrained_lanes

When a constrained vector length increases, previously inaccessible bits are set to zero. Possible values are: 0=never, 1=always, 2=if the register was written to while the vector length was constrained

Type

int

Default value

0x0

sve.combine_movprfx_and_destructive

Attempt to combine the execution of MOVPRFX and the destructively-encoded instruction that follows it

Type

bool

Default value

0x0

sve.disable_speculative_accesses

All speculative memory accesses behave as though faulting, without accessing memory

Type

bool

Default value

0x0

sve.enable_at_reset

Start with system registers set up for Scalable Vector Extension use

Type

bool

Default value

0x0

sve.ffr_16b_pattern_UNKNOWN

A specific 16-bit UNKNOWN value that is used by parameter force_UNKNOWN_to_ffr.

Type

int

Default value

0x0

sve.force_UNKNOWN_to_ffr

Governs behavior if WRFFR writes a non-monotonic value to FFR. Possible values are: 0 - Write non-canonical value to FFR, 1 - Overwrite FFR with a specific pattern of 16-bit UNKNOWN value. See ffr_16b_pattern_UNKNOWN, 2 - Clear all bits above first zero 3 - Set all bits after first one

Type

int

Default value

0x0

sve.fp_exception_report_lowest

If true, for multiple trapped FP exceptions, report the lowest lane in VECITR. Otherwise, report the highest

Type

bool

Default value

0x0

sve.fp_exception_set_tfv

Set ESR_ELx.TFV during FP exception. Trapped exception flags are valid

Type

bool

Default value

0x1

sve.fp_exception_set_vecitr

If true, set ESR_ELx.VECITR during FP exception. Otherwise, set RES0.

Type

bool

Default value

0x0

sve.has_b16b16

Whether FEAT_SVE_B16B16 is implemented. Possible values are: 0 - Not implemented, 1 - Implemented if either FEAT_SVE2 or FEAT_SME2 is implemented

Type

int

Default value

0x0

sve.has_sme

Whether SME is implemented (FEAT_SME)

Type

bool

Default value

0x0

sve.has_sme2

Whether SME2 is implemented (FEAT_SME2)

Type

bool

Default value

0x0

sve.has_sme_f16f16

Whether FEAT_SME_F16F16 is implemented. Possible values are: 0 - Not implemented, 1 - Implemented if FEAT_SME2 is implemented

Type

int

Default value

0x0

sve.has_sme_f64f64

If SME is implemented, whether double-precision FMOPA and FMOPS are implemented

Type

int

Default value

0x1

sve.has_sme_f8f16

If SME2 is implemented, whether FEAT_SME_F8F16 is implemented

Type

int

Default value

0x1

sve.has_sme_f8f32

If SME2 is implemented, whether FEAT_SME_F8F32 is implemented

Type

int

Default value

0x1

sve.has_sme_fa64

Whether FEAT_SME_FA64 is implemented

Type

bool

Default value

0x0

sve.has_sme_i16i64

If SME is implemented, whether instructions that accumulate 16-bit integer outer products into 64-bit integer tiles are implemented

Type

int

Default value

0x1

sve.has_sme_lutv2

Whether FEAT_SME_LUTv2 is implemented

Type

bool

Default value

0x0

sve.has_sme_priority_control

Whether SME Priority Control is implemented

Type

bool

Default value

0x1

sve.has_sve2

Whether SVE2 is implemented (FEAT_SVE2).

Type

bool

Default value

0x0

sve.has_sve2_aes

If SVE2 is implemented, whether AES instructions are implemented. Possible values are:
0 - not implemented, 1 - SVE2 AESE, AESD, AESMC, and AESIMC are implemented (FEAT_SVE_AES), 2 - Same as 1 but in addition SVE2 PMULLB and PMULLT with 64-bit source are implemented (FEAT_SVE_PMULL128).

Type

int

Default value

0x2

sve.has_sve2_bit_perm

If SVE2 is implemented, whether BitPerm instructions are implemented (FEAT_SVE_BitPerm).

Type

bool

Default value

0x1

sve.has_sve2_sha3

If SVE2 is implemented, whether SHA3 instructions are implemented (FEAT_SVE_SHA3).

Type

bool

Default value

0x1

sve.has_sve2_sm4

If SVE2 is implemented, whether SM4 instructions are implemented (FEAT_SVE_SM4).

Type

bool

Default value

0x1

sve.has_sve_bf16

Whether SVE BFloat16 instructions are implemented

Type

bool

Default value

0x1

sve.has_sve_extended_bf16

Deprecated: to enable FEAT_EBF16, use CPU parameter has_ebf16. Whether Extended BFloat16 instructions are implemented. Possible values are: 0 - Disabled, 1 - Enabled if SME or SVE is implemented, 2 - Enabled if SME is implemented

Type

int

Default value

0x2

sve.has_sve_mm_f32

Whether the SVE FP32 Matrix Multiply instructions are implemented (FEAT_F32MM).

Type

bool

Default value

0x1

sve.has_sve_mm_f64

Whether the SVE FP64 Matrix Multiply instructions are implemented (FEAT_F64MM).

Type

bool

Default value

0x1

sve.has_sve_mm_i8

Whether the SVE Int8 Matrix Multiply instructions are implemented (FEAT_I8MM).

Type

bool

Default value

0x1

sve.movprfx_unpredictable_behavior

Defines the behavior of MOVPRFX and the instruction it immediately precedes when the behavior is CONSTRAINED UNPREDICTABLE. Possible values are: 0 - UNDEF execution from MOVPRFX, 1 - MOVPRFX and second half of instruction executes as NOP, 2 - NOP MOVPRFX only, 3 - UNDEF execution from MOVPRFX unless otherwise trapped

Type

int

Default value

0x0

sve.predicated_sp_align_check_behaviour

Governs behavior of SP alignment checking for predicated memory accesses. Possible values are: 0 - Always perform, 1 - Skip if governing predicate is 0, 2 - Skip for contiguous accesses if governing predicate is 0, 3 - Skip for gather/scatter accesses if governing predicate is 0

Type

int

Default value

0x0

sve.relax_sme_watchpoint_matching_16

Whether memory accesses through Z and P registers in Streaming Mode and all accesses through ZA match watchpoints rounded to 16-byte alignment

Type

bool

Default value

0x0

sve.relax_sve_watchpoint_matching_16

If FEAT_DEBUGv8p9 is implemented, whether memory accesses through Z and P registers outside Streaming Mode match watchpoints rounded to 16-byte alignment

Type

bool

Default value

0x0

sve.sm_tag_checked

Whether SME, SVE, and SIMD&FP load and store instructions executed when the PE is in Streaming SVE mode perform a Tag Check

Type

bool

Default value

0x1

sve.sme2_version

The version of SME2 if implemented. Possible values are: 0 - FEAT_SME2, 1 - FEAT_SME2p1

Type

int

Default value

0x0

sve.sme_only

If SME is implemented, whether SVE functionality is available only when SM=1

Type

bool

Default value

0x0

sve.sme_ssve_fp8_support_level

If FEAT_SME2 and FEAT_FP8 are implemented, whether FP8 operations are supported in Streaming Mode where not implemented outside Streaming Mode. Possible values are: 0 - No support above FEAT_FP8, 1 - FEAT_SSVE_F8FMA, 2 - FEAT_SSVE_F8DOT4, 3 - FEAT_SSVE_F8DOT2

Type

int

Default value

0x0

sve.sme_veclens_implemented

Which SME vector lengths are implemented. Represented as a bitfield where bit[n]==1 implies SME vector length of $128 \cdot 2^n$ bits is implemented

Type

int

Default value

0x7

sve.smidr_el1_implementer_val

The value of SMIDR_EL1.Implementer

Type

int

Default value

0x41

sve.smidr_el1_revision_val

The value of SMIDR_EL1.Revision

Type

int

Default value

0x0

sve.sve2_version

The version of SVE2 if implemented. Possible values are: 0 - FEAT_SVE2, 1 - FEAT_SVE2p1

Type

int

Default value

0x0

sve.sve_dabt_far_behaviour

Whether the FAR reported on a Data Abort is imprecise. Possible values are: 0 - FAR Precise, 1 - FAR not Precise on abort due to an SVE contiguous vector load/store in Streaming SVE mode, or an SME load/store. , 2 - As per 1, but only for predicated SVE/SME instructions.

Type

int

Default value

0x0

sve.sve_wp_far_behaviour

FAR reporting behavior on a Watchpoint debug exception. Possible values are: 0 - FAR Precise, 1 - FAR not Precise on abort due to an SVE contiguous vector load/store in Streaming SVE mode, or an SME load/store, 2 - FAR not valid on abort due to an SVE contiguous vector load/store in Streaming SVE mode, or an SME load/store, 3 - As per 1, but only for predicated SVE/SME instructions.

Type

int

Default value

0x0

sve.trace_za_tilewise

Whether tile-wise accesses to ZA are traced tile-wise rather than array-wise. Note: if false, column-wise accesses cause an event for every vector in the tile

Type

bool

Default value

0x1

sve.undef_invalid_combined_movprfx

If a combined MOVPRFX is invalid, raise an UNDEF exception. Otherwise, NOP the second half. This parameter is deprecated

Type

bool

Default value

0x1

sve.unknown_value

Simulated value for a state that has an UNKNOWN value after reset

Type

int

Default value

0xdeaddeaddeaddead

sve.vecLEN

SVE vector length in units of 64 bits

Type

int

Default value

0x8

sve.z_reg_on_load_fault_behaviour

Governs the behavior of destination Z-registers in case of a load fault. Possible values are: 0 - Register becomes UNKNOWN, 1 - Register is preserved

Type

int

Default value

0x0

sve.za_on_svl_increase_behaviour

Controls the state of the previously inaccessible portion of the ZA registers on SVL increase. Possible values are: 0 - Retain values, 1 - Zero ZA

Type

int

Default value

0x0

sve.za_tag_checked

Whether memory accesses due to SME LDR and STR instructions that access the SME ZA array perform a Tag Check

Type

bool

Default value

0x1

swp_with_xzr_is_st_atomic

If true, swp with dest as xzr is treated as store atomic

Type

bool

Default value

0x1

sync_ext_abort_is_sync_error

Treat synchronous external aborts as synchronous SErrors (RASv8.9). 0, synchronous external abort. 1, synchronous error.

Type

bool

Default value

0x0

system_pmu_id

When FEAT_SPMU is implemented, indicates the largest value *s* to select a System PMU <*s*>

Type

int

Default value

0x0

take_ccfail_tsc_trap

When take_ccfail_undef=1 this parameter controls whether or not an SMC instruction that is trapped by HCR_EL2.TSC but fails its condition code check generates a trap to EL2.

Type

bool

Default value

0x0

take_ccfail_undef

UNDEF exception is taken even if condition code check fails

Type

bool

Default value

0x1

tcr_ps_reserved_value_size

Physical size treated when TCR.(I)PS is programmed with a reserved value. 0, 48 bits. 1, 52 bits. The parameter value is treated 0 if LPA is not supported.

Type

int

Default value

0x0

tcr_txsz_undersize_should_fault

If large VA is not supported, Whether undersized TxSZ value should generate translation fault

Type

bool

Default value

0x1

tdosa_traps_osdlr_if_no_os_double_lock

MDCR_EL*.TDOSA enables trap on OSDLR_EL1 and DBGOSDLR when OS double-lock is not implemented.

Type

bool

Default value

0x1

tidcp_traps_el0_undef_imp_def

TIDCP has priority over UNDEF for accesses to IMPLEMENTATION DEFINED functionality from EL0

Type

bool

Default value

0x1

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_or_ic_invalid_xt

Behavior of TLBI and IC instructions that don't take Xt as an argument when Xt != 0b11111.
0: TLBI and IC not UNDEF, 1: TLBI UNDEF, IC not UNDEF, 2: TLBI not UNDEF, IC UNDEF, 3: TLBI and IC UNDEF

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

tme_disable-read-write-set-optimizations

If true, disables the read/write set related optimizations of the model.

Type

bool

Default value

0x0

tme_imp-failures-can-retry

If true, IMP=1 failures introduced by the parameters: tme_wfe-fails-transactions tme_tcommit-fails-transactions tme_wakeup-from-wfe-always-fails-transactions will also report RTRY=1.

Type

bool

Default value

0x0

tme_implementation-type

Implementation type for TME.

The following options are available: 0x0: Always fail starting transactions with the IMP cause. 0x1: Fail on forbidden operations (e.g. some system register accesses) and at model's convenience. 0x2: As 0x1 but also enable inter PE memory conflict checking.

Type

int

Default value

0x0

tme_random-memory-access-fail-chance

If >0, add a pseudorandom chance for every memory access (loads, stores, TCOMMIT) inside a transaction to cause the transaction to fail with IMP.

Type

int

Default value

0x0

tme_read-set-size

Size of the transactional read set in bytes, rounded up to the nearest integer number of transaction granules. 0 == unlimited

Type

int

Default value

0x0

tme_support-only-guaranteed-mem-attr

If true, a transactional access to memory with a type not architecturally guaranteed to be supported will cause a transaction failure with IMP=1

Type

bool

Default value

0x0

tme_tcommit-fails-transactions

If true, executing TCOMMIT inside a transaction will cause it to fail with IMP=1

Type

bool

Default value

0x0

tme_wakeup-from-wfe-always-fails-transactions

If true, waking up from a WFE will always fail the transaction, even if not required

Type

bool

Default value

0x0

tme_wfe-fails-transactions

If true, executing WFE inside a transaction will cause it to fail with IMP=1

Type

bool

Default value

0x0

tme_write-set-size

Size of the transactional write set in bytes, rounded up to the nearest integer number of transaction granules. 0 == unlimited

Type

int

Default value

0x0

trace_full_simd_reg_with_nep

Whether full simd register is traced even if partial update is done when FPCR.NEP=1

Type

bool

Default value

0x0

trace_has_sysreg_access

ETM trace registers support access via system registers

Type

bool

Default value

0x1

trace_icc_registers_as_icv_when_redirected

If true, update trace with ICV, instead of ICC when ICV registers are accessed depending on the core state.

Type

bool

Default value

0x0

trace_physical_registers_when_host_virtualisation_enabled

When host virtualisation is enabled, trace sysreg accesses to physical register accessed (0=disabled, 1=Trace only ELR/SPSR_EL1 as ELR/SPSR_EL2, 2=Trace all redirected registers as physical registers)

Type

int

Default value

0x0

trap_dc_cmo_to_pou_if_nop

Whether traps to DC CMO operations to PoU are ignored if the same is treated as NOP.

Type

bool

Default value

0x1

trap_ic_cmo_to_pou_if_nop

Whether traps to IC CMO operations to PoU are ignored if the same is treated as NOP.

Type

bool

Default value

0x1

trap_reserved_group3_id_regs

Whether setting HCR_EL2.TID3 traps reserved group3 id registers.

Type

bool

Default value

0x0

trbe_cmod

TRBE Customer Modified.

Type

int

Default value

0x0

trbe_des

Designer, JEP106 identification code.

Type

int

Default value

0x0

trbe_external_abort_handling

Describes how the PE manages External aborts on writes made by the Trace Buffer Unit to the trace buffer. (0->External abort is reported to TRBE. From Armv9.3, the value 0 is not

permitted and will be 1 if Armv9.3 is implemented. 1-> External abort is ignored. 2->The External abort generates an SError and the error is not reported to TRBE.)

Type

int

Default value

0x0

trbe_has_hardware_translation_table_update

If true, address translation performed by the Trace Buffer Extension manages the Access Flag and dirty state

Type

bool

Default value

0x1

trbe_implemented_version

Trace Buffer Extension implemented version, 1: FEAT_TRBE implemented (Armv9.0).

Type

int

Default value

0x1

trbe_mpam

TRBE MPAM support.

Type

int

Default value

0x0

trbe_part

Part number.

Type

int

Default value

0x0

trbe_partid_max

Largest permitted TRBDEVID1.PARTID value.

Type

int

Default value

0x0

trbe_pmg_max

Largest permitted TRBDEVID1.PMG value.

Type

int

Default value

0x0

trbe_revand

TRBE component minor revision.

Type

int

Default value

0x0

trbe_revision

TRBE architecture revision.

Type

int

Default value

0x0

treat-dcache-cmos-to-poc-as-nop

Whether dcache maintenance operations to the point of coherency are required for instruction to data coherence. 0 - Clean/Invalidate ops required, 1 - Clean/Invalidate ops not required and cannot generate faults, 2 - Clean/Invalidate ops not required but can generate faults

Type

int

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

treat-dcache-invalidate-as-clean-invalidate

Treat data cache invalidate operations as clean and invalidate.

Type

bool

Default value

0x0

treat-icache-cmos-to-pou-as-nop

If has_coherent_icache is true, whether instruction cache invalidation operations to PoU which are treated as NOP can generate fault. 0 - cannot generate faults, 1 - can generate faults

Type

int

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

treat_forced_normal_as_device_for_excl_atomics

Whether exclusive/atomic access is supported in same manner as access to device if stage1 is Device memory and final memory attribute forced to normal by FWB

Type

bool

Default value

0x0

treat_pld_as_nop

If true, treat PLD as NOP.

Type

bool

Default value

0x0

treat_pli_as_nop

If true, treat PLI as NOP.

Type

bool

Default value

0x0

treat_wfi_wfe_as_nop

If true, never go into wait state for WFI or WFE instructions.

Type

bool

Default value

0x0

truncate_pc_on_illegal_exception_return_to_aarch32

On Illegal ERET to AArch32, truncate PC to 32-bits

Type

bool

Default value

0x1

undef_ccsidr2_access_for_unimplemented_aarch32

Whether access to CCSIDR2 is undef if AArch32 is implemented or not at EL1

Type

bool

Default value

0x0

unification-level

Level of Unification Inner Shareable for the cache hierarchy

Type

int

Default value

0x1

unification-uniprocessor-level

Level of Unification Uniprocessor for the cache hierarchy

Type

int

Default value

0x1

unpred_LSE128_overlap

Constrained unpredictable behaviours for 128-bit LSE overlap. 1 Constraint_UNDEF, 2 Constraint_NOP

Type

int

Default value

0x1

unpred_brb_iall_or_inj_invalid_xt_behave_as_undef

If true, BRB IALL/INJ instruction will behave as UNDEFINED if Xt != 0b11111

Type

bool

Default value

0x0

unpred_clear_ISV_for_exception_before_software_step

Whether ESR_ELx.ISV bit is cleared/set, when it is constrained unpredictable due to a different exception before a software step exception.

Type

bool

Default value

0x0

unpred_edscr_ns_set_unknown_bit

Unknown(x) bit in NS field in EDSCR can be configure to 0 or 1.

Type

bool

Default value

0x0

unpred_edscr_rw_unknown_bits_read_as_1

Unknown(x) bits in RW field in EDSCR are read as 1 instead of 0.

Type

bool

Default value

0x0

unpred_edscr_status_read_as_no_syndrome

Controls the choice of EDSCR.STATUS bit-values, when it is constrained unpredictable behaviour due to a different exception before a halting step debug event.

Type

bool

Default value

0x0

unpred_extdbg_unknown_bits

Data used to fill only in UNKNOWN bit-fields of external debug registers e.g., EDPFR and EDDFR.

Type

int

Default value

0x0

unpred_load_single_reg_overlap_with_wb

Constrained unpredictable behaviours for single load with writeback(might impact certain load pair instructions) 0 Constraint_WBSUPPRESS, 1 Constraint_UNDEF, 2 Constraint_NOP

Type

int

Default value

0x0

unpred_mrsmsr_currentlymapped_undef

UNPREDICTABLE register access (accessible from current mode using different instruction) modeled as NOP when false and UNDEF when true

Type

bool

Default value

0x0

unpred_mrsmsr_protfailed_undef

UNPREDICTABLE register access (not accessible from current PL and security state) modeled as NOP when false and UNDEF when true

Type

bool

Default value

0x0

unpred_mte_stzgm_tag_operation_before_data

Whether Tag operations are performed before data operations for an STGZM instruction.

Type

bool

Default value

0x1

unpred_mte_tag_read_when_ata_controls_are_zero_or_untagged_attr

Constrained unpredictable for MTE tag read when ATA controls are 0 or untagged attribute. false, Read as zero. true, Permitted to generate an external abort if a read of data from the same address would generate an external abort.

Type

bool

Default value

0x0

unpred_mte_tag_store_data_cache_instr_to_device_mem_as_alignment_fault

Constrained unpredictable choice for MTE instructions which store tags (on DC instructions) to memory locations marked as Device. false, Storing the data, if any, to the locations. true, Generating an Alignment Fault.

Type

bool

Default value

0x0

unpred_mte_tag_store_to_device_mem_as_alignment_fault

Constrained unpredictable choice for STZGM instruction which store tags to memory locations marked as Device. false, Storing the data, if any, to the locations. true, Generating an Alignment Fault.

Type

bool

Default value

0x0

unpred_nested_virtualization_nv_behaviour

Constrained unpredictable choices for HCR_EL2.NV=0 and HCR_EL2.NV1=1 with respect to nested virtualization - 0, Behave as defined in the specification as per bit values - 1, Behave as if HCR_EL2.NV=1 and HCR_EL2.NV1=1 for all purpose other than reading back HCR_EL2.NV - 2, Behave as if HCR_EL2.NV=0 and HCR_EL2.NV1=0 for all purpose other than reading back HCR_EL2.NV1

Type

int

Default value

0x0

unpred_par_attr_returns_mair

If true, PAR_EL1.ATTR represents the memory attributes as per the MAIR value instead of the ones in the descriptor.

Type

bool

Default value

0x0

unpred_sctlr_c_0_taggable_behaviour

Controls unpredictable effects when SCTLTR_ELx.C=0 for a stage 1 translation regime on whether memory is treated as taggable. Values: 0=Tagged, 1=Untagged but forced to Tagged when FWB=1 and stage 2 restores WB, 2 = Untagged

Type

int

Default value

0x2

unpred_store_exclusive_base_overlap

Constrained unpredictable behaviours for store exclusive when s==n. 0 Constraint_NONE, 1 Constraint_UNDEF, 2 Constraint_NOP

Type

int

Default value

0x0

unpred_store_pair_and_single_reg_overlap_with_wb

Constrained unpredictable behaviours for pair and single store with writeback(doesn't cover store exclusive) 0 Constraint_NONE, 1 Constraint_UNDEF, 2 Constraint_NOP

Type

int

Default value

0x0

unpred_tlbis_not_in_monitor_mode

Constrained unpredictable behaviors for AArch32 TLBI instructions executed in secure privileged mode other than Monitor mode. 0: Preferred behavior (default), 1: UNDEF, 2: NOP, 3: execute as if had been executed in Monitor mode

Type

int

Default value

0x0

unpred_tsize_aborts

Behaviour when TSize is out of range. 0, force into range. 1, translation fault, forces unpred_tsize_pamax_aborts to 1.

Type

bool

Default value

0x0

unpred_tsize_pamax_aborts

Behaviour when stage 2 TSize exceeds the physical address size (or 40bits, from AArch32). 0, force into range. 1, translation fault. Ignored if unpred_tsize_aborts is 1.

Type

bool

Default value

0x0

unpred_vnchr_el2_ress_mismatch

Constrained unpredictable choices when bits marked as RESS do not all have the same value for VNCR_EL2 - 0, Generating an EL2 translation regime translation abort on use of the VNCR_EL2 register - 1, Reserved sign extended bits of VNCR_EL2 are same as bit[52] or bit[48] based on if large VA is supported or not, for all purposes other than reading back the register - 2, Reserved sign extended bits of VNCR_EL2 are same as bit[52] or bit[48] based on if large VA is supported or not, for all purposes

Type

int

Default value

0x0

unpred_zero_spsr_btype

Constrained unpredictable control to make SPSR_ELx.BTYPE 0 instead of PSTATE.BTYPE on synchronous exceptions other than Software Step, PC alignment fault, Instruction Abort, Breakpoint or Address Matching Vector Catch, Illegal Execution State, BRK instruction, Branch Target

Type

bool

Default value

0x1

unpredictable_exclusive_abort_memtype

Cause MMU abort if exclusive access is not supported in certain memory type (0=exclusives allowed in all memory types, 1=exclusives abort in Device memory types, 2=exclusives abort in any type other than WB inner cacheable)

Type

int

Default value

0x0

unpredictable_hvc_behaviour

HVC unpredictable behaviour. 0, UNDEF. 1, NOP.

Type

int

Default value

0x0

unpredictable_smc_behaviour

SMC unpredictable behaviour. 0, UNDEF. 1, NOP.

Type

int

Default value

0x0

unpredictable_wfet_and_wfit_behaviour

WFET and WFIT unpredictable behaviour in debug state. 0, UNDEFINED. 1, NOP.

Type

int

Default value

0x1

unsupported_atomic_fault_type

Type of fault reported on unsupported atomic access. 0 = external abort if any reported by interconnect, 1 = precise unsupported atomic fault, 2 = precise external abort, 3 = imprecise external abort.

Type

int

Default value

0x0

unsupported_hw_update_fault_type

Type of abort reported when hw update to descriptor is done using unsupported memtype (0=No abort, 1=IMPDEF abort caused by memtype, 2=Sync external abort)

Type

int

Default value

0x0

use_Xt_as_LDG_STG_input

Use new MTE Instructions formats for LDG/ST(Z)(2)G, which use Xt as an input

Type

bool

Default value

0x1

use_architectural_names

Use names SP/LR/PC instead of R13/R14/R15

Type

bool

Default value

0x0

use_mte_eac_02_instructions_encoding

Use new MTE Instructions encoding since MTE spec EAC 0.2 in ARMv8.5.

Type

bool

Default value

0x1

use_mte_eac_08_tfsr_encoding

Use new MTE TFSR_ELx encodings since MTE spec EAC 0.8 in ARMv8.5.

Type

bool

Default value

0x1

use_rosetta_disass

Use Rosetta disassembly library.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

use_sif_to_compute_pan

Where FEAT_PAN3 is implemented, whether SCR_EL3.SIF bit is used to determine instruction access permission for the purpose of PAN

Type

bool

Default value

0x0

use_stage1_sh_as_input_to_stage2

IMPDEF case of whether to use stage1 shareability or OuterShareable as input to stage2 if stage1 is Device memtype

Type

bool

Default value

0x0

use_tlb_contig_hint

Translation table entries with the contiguous hint bit set generate large TLB entries.

Type

bool

Default value

0x0

user_defined_rom_table_debug_power_config

User defined ROM Table debug power domains for ED,CTI,PMU and TRACE, and DBGPCR configuration. The "version" field and "cores" array are mandatory. The "dbgpcr" array, if provided, must contain unique integers in the range [0, 32) describing which debug power domains have power control implemented. The "rom" and "dbgpcr" fields in objects in the "cores" array are only allowed when 'debug_rom_is_flat' is false. All power domain ID fields ("rom", "ed/pmu", "cti", "etm") must be in the range [0, 32). The "ed/pmu" field is mandatory. Example JSON for a hierarchical debug ROM layout: '{"version": 0, "dbgpcr": [0, 1], "cores": [{"dbgpcr": [1, 31], "rom": 0, "ed/pmu": 0, "cti": 31, "etm": 1}, {"ed/pmu": 0}]}'

Type

string

Default value**vpu_datapath_width**

VPU data path width

Type

int

Default value

0x80

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

warn_unpredictable_in_v7

If true, behaviour which is unpredictable in V7 yet is predictable in V8 will produce a warning

Type

bool

Default value

0x1

watchpoint-log2secondary_restriction

log2 size of secondary restriction of FAR/EDWAR possible values on watchpoint hit for load/store operations.

Type

int

Default value

0x0

wfe_wakeup_delay

Configure WFE wakeup delay in CPU cycles

Type

int

Default value

0x0

wfi_wakeup_delay

Configure WFI wakeup delay in CPU cycles

Type

int

Default value

0x0

wnr_is_read_for_s2f_on_s1_atomic_instr_fault

Whether WnR is 0 for stage2 fault on stage1 for atomic instructions

Type

bool

Default value

0x0

wnr_is_read_for_s2f_on_s1_dbm_update

Whether WnR is 0 for stage2 fault on stage1 descriptor dbm update

Type

bool

Default value

0x0

wp_ignores_dbm_update

If true, dbm update is ignored on watchpoint hit

Type

bool

Default value

0x0

wp_num_reporting

When reporting of the watchpoint number on Watchpoint Exceptions and Debug Events is performed 0 - When FEAT_Debugv8p9 is implemented or otherwise required 1 - When FEAT_Debugv8p9 or FEAT_SME is implemented

Type

int

Default value

0x0

3.5.3 ARMAEMv8MCT

ARMAEMv8MCT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-182: IP revisions support

Revision	Quality level
v8.0M	Full support
v8.1M	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMAEMv8MCT

This model has the following Iris instances:

Table 3-183: ARMAEMv8MCT Iris instances

InstanceName	ComponentName
ARMAEMv8MCT	ARM_AEMv8M
ARMAEMv8MCT.acp_mapper	PVBusMapper
ARMAEMv8MCT.ext_bus	PVBusLogger
ARMAEMv8MCT.ext_bus.mapper	PVBusMapper
ARMAEMv8MCT.l1_incoherent_interconnect	PVCache
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMAEMv8MCT.l1dcache	PVCache
ARMAEMv8MCT.l1dcache.upstream[0]	PVBusSlave
ARMAEMv8MCT.l1icache	PVCache
ARMAEMv8MCT.l1icache.upstream[0]	PVBusSlave
ARMAEMv8MCT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-184: ARMAEMv8MCT MTI instances

InstanceName	ComponentName
ARMAEMv8MCT	ARM_AEMv8M
ARMAEMv8MCT.acp_mapper	PVBusMapper
ARMAEMv8MCT.ext_bus	PVBusLogger
ARMAEMv8MCT.ext_bus.mapper	PVBusMapper
ARMAEMv8MCT.l1_incoherent_interconnect	PVCache
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMAEMv8MCT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMAEMv8MCT.l1dcache	PVCache
ARMAEMv8MCT.l1dcache.upstream[0]	PVBusSlave
ARMAEMv8MCT.l1icache	PVCache
ARMAEMv8MCT.l1icache.upstream[0]	PVBusSlave
ARMAEMv8MCT.l2_flusher	AsyncCacheFlushUnit

ARMAEMv8MCT contains the following CADI targets:

- ARM_AEMv8M

Ports for ARMAEMv8MCT

Table 3-185: Ports

Name	Protocol	Type	Description
ahbd	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
ahbp_m	PVBus	Master	The core will generate Vendor System data accesses on this port.
ahbs	PVBus	Slave	External master (e.g. DMA) can write TCMs (whether or not enabled in xTCMCR).
auxfault	Value	Slave	This is wired to the Auxiliary Fault Status Register.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
coproc_bus	CoprocBusProtocol	Slave	Co-Processor Interface
cpuwait	Signal	Slave	When this signal is HIGH out of reset, it forces the processor into a quiescent state that delays its boot-up sequence and instruction execution until this signal is driven LOW.

Name	Protocol	Type	Description
currpri	Value	Master	Current execution priority.
dap_s	PVBus	Slave	Debug Access Port (DAP).
dbgen	Signal	Slave	Invasive debug enable.
dbgrestart	Signal	Slave	External debug request.
dbgrestarted	Signal	Master	External debug request.
edbgrq	Signal	Slave	External debug request.
etm_reset	Signal	Slave	Separate reset for ETM, if param "has_etm_reset" is true.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
fpxxc	Value	Master	Port that sends the value of the FPxxC exception flags (FPIXC, FPIDC, FPOFC, FPUFC, FPDZC, FPIOC). Each flag is present in the same position as in the FPSCR register (e.g. fpxxc[0] = FPIOC).
halted	Signal	Master	External debug request.
idau	PVBus	Master	The core will generate IDAU Bus request.
idau_invalidate_region	Value_64	Slave	64 bit number to invalid IDAU memory range (start_address<<32 end_address)
initpahben	Signal	Slave	Enable P-AHB on the next reset
initvtor_ns	Value	Slave	Reset configuration port - Non-Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset This port remains functional no matter ARMv8-M Security Extensions are included or not When ARMv8-M Security Extensions are not included, all exceptions will use NS vector base address given by this port.
initvtor_s	Value	Slave	Reset configuration port - Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset It becomes functional when ARMv8-M Security Extensions are included When ARMv8-M Security Extensions are not included, this port will be ignored.
intisr[496]	Signal	Slave	This signal array delivers signals to the NVIC.
intnmi	Signal	Slave	Configure non maskable interrupt.
intnum	Value	Master	Exception number of the current execution context (from IPSR[8:0]) When the processor is in Thread mode, INTNUM is 0 When the processor is in Handler mode, INTNUM is the exception number of the currently-executing exception.
io_port_in	PVBus	Slave	I/O port pair. See the documentation for the io_port_out port.
io_port_out	PVBus	Master	I/O port pair. Used if IOP is true. Transactions from io_port_out which do not "match" should be returned via io_port_in. For performance reasons, the I/O port interface is not modelled directly. Instead, a simple PVBus gasket is inserted at the point in the memory system where the I/O port would be. In hardware, a device would be attached to the port which would tell the CPU whether it would like to intercept each transaction, given its address. This can be modelled in a performant manner by connecting a PVBusMapper-based device to io_port_out which intercepts transactions of interest and passes other transactions back to the CPU via io_port_in. Your I/O port device model is also responsible for aborting transactions which would be aborted on hardware (e.g. exclusives) if necessary.
lockdcaic	Signal	Slave	-
locknsmpu	Signal	Slave	Disable writes to the Non-Secure MPU_*_NS registers
locknsvtor	Signal	Slave	Disable writes to VTOR_NS

Name	Protocol	Type	Description
lockpahb	Signal	Slave	P-AHB related ports Disable writes to PAHBCR
locksau	Signal	Slave	Disable writes to the SAU_* registers
locksmpu	Signal	Slave	Disable writes to the Secure MPU_* registers
locksvtaircr	Signal	Slave	Disable writes to VTOR_S, AIRCR.PRIS, AIRCR.BFHFNMINIS
lockup	Signal	Master	Asserted when the processor is in lockup state.
niden	Signal	Slave	Non-invasive debug enable.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbush_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
spiden	Signal	Slave	Secure invasive debug enable.
spniden	Signal	Slave	Secure non-invasive debug enable.
stcalib[2]	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.

Parameters for ARMAEMv8MCT

AFSR_type

0:No AFSR, 1:state-only 2:Many bits set from (im)precise aborts on AXI, TCM, etc

Type

int

Default value

0x1

AIRCR.BFHFNMINIS_reset

If true, set the bit after reset (as if by IMP_DEF mechanism). Ignored if SECEXT=false

Type

bool

Default value

0x0

AIRCR.BFHFNMINIS_writable

Is AIRCR.BFHFNMINIS bit[13] writeable

Type

bool

Default value

0x1

AIRCR.ENDIANNESS

Initialize processor to big endian mode

Type

bool

Default value

0x0

AIRCR.PRIS_writable

Is AIRCR.PRIS bit[14] writeable

Type

bool

Default value

0x1

AIRCR.VECTCLRACTIVE_changes_mode

Asserting AIRCR.VECTCLRACTIVE clears IPSR and any active exceptions. The mode is also changed to thread if this flag is true. Ignored for v8-M

Type

bool

Default value

0x1

AIRCR_NS.DIT_reset

If true and AIRCR_NS.DIT_writable==0, set the bit after reset (as if by IMP_DEF mechanism)

Type

bool

Default value

0x1

AIRCR_NS.DIT_writable

Is AIRCR_NS.DIT bit[4] writeable

Type

bool

Default value

0x1

AIRCR_S.DIT_reset

If true and AIRCR_S.DIT_writable==0, set the bit after reset (as if by IMP_DEF mechanism)

Type

bool

Default value

0x1

AIRCR_S.DIT_writable

Is AIRCR_S.DIT bit[4] writeable

Type

bool

Default value

0x1

BB_PRESENT

Enable bitbanding

Type

bool

Default value

0x0

BEATS_PER_TICK

Number of beats from each in-flight vector instruction executed in 1 tick (1,2 or 4).

Type

int

Default value

0x2

BF_is_nop

BF instruction executes as NOP, even if we have LO_BRANCH_INFO.

Type

bool

Default value

0x1

CCR.BP

Reset value of the Configuration and Control Register's branch prediction enable bit

Type

bool

Default value

0x1

CCR.BP_writable

Whether it is possible to modify the Configuration and Control Register's branch prediction enable bit

Type

bool

Default value

0x0

CDEMAPPEDONCP

Bit N specifies whether the instruction for coprocessor N (CP7:CP0) is redirected to the CDE module

Type

int

Default value

0xff

CDERTLID

Value of ID_AFR0.CDERTLID

Type

int

Default value

0x20

CFGMEMALIAS

Memory address alias bit for the ITCM, DTCM and P-AHB regions. 0=No alias, 1=Alias bit 24, 2=Alias bit 25, 4=Alias bit 26, 8=Alias bit 27, 16=Alias bit 28

Type

int

Default value

0x0

CFGNOCDECP

Bit N means external coprocessor N (CP7:CP0) disable for CDE coprocessor

Type

int

Default value

0x0

CFGPAHBSZ

Size of the P-AHB peripheral port memory region. 0=P-AHB disabled, 1=64MB, 2=128MB, 3=256MB, 4=512MB

Type

int

Default value

0x0

CPNSPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Non-Secure state

Type

int

Default value

0xff

CPSPPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Secure state

Type

int

Default value

0xff

CPUID

Set SCS CPUID Base Register. If set to zero, a default CPUID is used.

Type

int

Default value

0x0

CTI

CTI (Cross Trigger Interface) included

Type

bool

Default value

0x0

CTI_irq0_pin

CTI interrupt request 0 pin

Type

int

Default value

0x4

CTI_irq1_pin

CTI interrupt request 1 pin

Type

int

Default value

0x5

DTGU

DTCM Security Gate Unit included

Type

bool

Default value

0x0

DTGUBLKSZ

DTCM gate unit block size. Size=pow(2, DTGUBLKSZ + 5) bytes

Type

int

Default value

0x3

DTGUMAXBLKS

Maximum number of DTCM gate unit blocks. Number of blocks=pow(2, DTGUMAXBLKS)

Type

int

Default value

0x0

DWT_CTRL.NOCYCCNT

DWT cycle-counter not present (v8M_bl/v6M never have one).

Type

bool

Default value

0x0

DWT_CTRL.NOPRFCNT

DWT performance-counters not present (v8M_bl/v6M never have them).

Type

bool

Default value

0x0

DWT_CTRL.NUMCOMP

Number of watchpoint unit comparators implemented

Type

int

Default value

0x4

DWT_DEVARCH.REVISION

0: V2, 1: V2.1.

Type

int

Default value

0x1

DWT_FUNCTION0.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION0. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0xb

DWT_FUNCTION1.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION1. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0x1e

DWT_FUNCTION10.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION10. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0xb

DWT_FUNCTION11.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION11. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0x1e

DWT_FUNCTION12.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION12. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0xb

DWT_FUNCTION13.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION13. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0x1e

DWT_FUNCTION14.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION14. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0xb

DWT_FUNCTION15.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION15. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0x1e

DWT_FUNCTION2.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION2. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0xb

DWT_FUNCTION3.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION3.
If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0x1e

DWT_FUNCTION4.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION4.
If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0xb

DWT_FUNCTION5.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION5.
If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0x1e

DWT_FUNCTION6.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION6.
If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0xb

DWT_FUNCTION7.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION7.
If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0x1e

DWT_FUNCTION8.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION8. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0xb

DWT_FUNCTION9.ID

Sets the capabilities of the comparator that is accessible via the register, DWT_FUNCTION9. If 'baseline' is set, invalid ID bits are cleared

Type

int

Default value

0x1e

DWT_TRACE

Support for DWT trace, controls the DWT_CTRL.NOTRCPKT bit. false : No DWT trace included, true: DWT trace included

Type

bool

Default value

0x1

DWT_VMASK_reset_data

DWT_VMASK register reset value

Type

int

Default value

0x0

ERRDEVID.NUM

RAS: Number of implemented error record indexes, 0 to 56.

Type

int

Default value

0x38

FPB_HAS_LSR

FPB has LAR and LSR for software lock if mainline

Type

bool

Default value

0x1

FP_CTRL.NUM_CODE

Number of breakpoint unit comparators implemented (limited to 15 in V6M or baseline)

Type

int

Default value

0x8

FP_CTRL.NUM_LIT

How many Literals FPB supports remapping (ignored if baseline or TZM)

Type

int

Default value

0x0

FP_REMAP.RMPSPT

FPB supports remapping (ignored if baseline or SECEXT)

Type

bool

Default value

0x1

ID_DFR0.Debug_Model_M_profile

Set whether debug extensions are implemented

Type

bool

Default value

0x1

ID_ISAR0.CmpBranch

Support for Compare and Branch instructions. 1 = Supports CBNZ and CBZ instructions; 3 = Supports non-predicated low overhead looping (WLS, DLS, LE, and LC) and branch future (BF, BFX, BFL, BFLX, and BFCSEL) instructions.

Type

int

Default value

0x3

ID_ISAR0.coproc_instrs

Supported Coprocessor instructions 0: None 1: CDP, LDC, MCR, MRC, and STC instructions
2: As for 1, and CDP2, LDC2, MCR2, MRC2, and STC2 instructions 3: As for 2, and MCRR
and MRRC instructions 4: As for 2, and MCRR and MRRC instructions

Type

int

Default value

0x4

ID_ISAR1.extend_instrs

level of support for extend instructions

Type

int

Default value

0x2

ID_ISAR1.interwork_instrs

level of support for Interworking instructions

Type

int

Default value

0x2

ID_ISAR2.MultiAccessInt

level of support for interruptible multi-access instructions

Type

int

Default value

0x2

ID_ISAR2.multS_instrs

level of support for advanced signed Multiply instructions

Type

int

Default value

0x3

ID_ISAR2.multU_instrs

level of support for advanced unsigned Multiply instructions

Type

int

Default value

0x2

ID_ISAR3.SIMD_instrs

level of support for SIMD instructions

Type

int

Default value

0x3

ID_ISAR3.saturate_instrs

level of support for saturate instructions

Type

int

Default value

0x1

ID_ISAR3.synchprim_instrs

level of support for synchronization primitives ID_ISAR3

Type

int

Default value

0x1

ID_ISAR4.synchPrim_instrs_frac

level of support for synchronization primitives ID_ISAR4

Type

int

Default value

0x3

ID_ISAR4.unpriv_instrssupported unprivileged instructions 0: None 1: LDRBT, LDRT, STRBT, and STRT instructions
2: As for 1, and LDRHT, LDRSBT, LDRSHT, and STRHT instructions**Type**

int

Default value

0x2

ID_ISAR4.withshifts_instrs

level of support for instructions with shifts

Type

int

Default value

0x3

ID_ISAR5.PACBTI

0: PAC/BTI not implemented, 1: PAC implemented using the QARMA5 algorithm with BTI, 2: PAC implemented using an IMP DEF algorithm with BTI, 4: PAC implemented using the QARMA3 algorithm with BTI

Type

int

Default value

0x0

ID_MMFR0.Auxiliary_registers

Auxiliary registers bits in ID_MMFR0, indicate the support for Auxiliary registers

Type

bool

Default value

0x1

ID_MMFR0.Outermost_shareability

Outermost shareability bits in ID_MMFR0, indicate the outermost shareability domain implemented

Type

int

Default value

0x0

ID_MMFR0.ShareLvl

Shareability levels bits in ID_MMFR0, indicate the number of Shareability levels implemented

Type

int

Default value

0x1

INITPAHBEN

The P-AHB enable state at reset

Type

bool

Default value

0x0

INITVTOR_NS

Non-Secure vector-table offset at reset

Type

int

Default value

0x0

INITVTOR_S

Secure vector-table offset at reset

Type

int

Default value

0x0

IOP

Send all d-side transactions to the port, io_port_out. Transactions which do not match should be returned to the port, io_port_in

Type

bool

Default value

0x0

IRQDIS0

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+0]

Type

int

Default value

0x0

IRQDIS1

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+32]

Type

int

Default value

0x0

IRQDIS10

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+320]

Type

int

Default value

0x0

IRQDIS11

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+352]

Type

int

Default value

0x0

IRQDIS12

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+384]

Type

int

Default value

0x0

IRQDIS13

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+416]

Type

int

Default value

0x0

IRQDIS14

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+448]

Type

int

Default value

0x0

IRQDIS15

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+480]

Type

int

Default value

0x0

IRQDIS2

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+64]

Type

int

Default value

0x0

IRQDIS3

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+96]

Type

int

Default value

0x0

IRQDIS4

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+128]

Type

int

Default value

0x0

IRQDIS5

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+160]

Type

int

Default value

0x0

IRQDIS6

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+192]

Type

int

Default value

0x0

IRQDIS7

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+224]

Type

int

Default value

0x0

IRQDIS8

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+256]

Type

int

Default value

0x0

IRQDIS9

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+288]

Type

int

Default value

0x0

ITGU

ITCM Security Gate Unit included

Type

bool

Default value

0x0

ITGUBLKSZ

ITCM gate unit block size. Size=pow(2, ITGUBLKSZ + 5) bytes

Type

int

Default value

0x3

ITGUMAXBLKS

Maximum number of ITCM gate unit blocks. Number of blocks=pow(2, ITGUMAXBLKS)

Type

int

Default value

0x0

ITM

Level of instrumentation trace supported. false : No ITM trace included, true: ITM trace included (unless baseline)

Type

bool

Default value

0x1

ITM_HAS_LSR

ITM support LAR and LSR for software lock

Type

bool

Default value

0x1

LOCKDTGU

Lock down of Data TGU registers write

Type

bool

Default value

0x0

LOCKITGU

Lock down of Instruction TGU registers write

Type

bool

Default value

0x0

LOCKTCM

Lock down of TCM registers write

Type

bool

Default value

0x0

LOCK_NS_MPU

Lock down of Non-Secure MPU registers write

Type

bool

Default value

0x0

LOCK_SAU

Lock down of SAU registers write

Type

bool

Default value

0x0

LOCK_S_MPU

Lock down of Secure MPU registers write

Type

bool

Default value

0x0

LVL_WIDTH

Number of bits of interrupt priority (baseline has 2)

Type

int

Default value

0x3

MEMORY_REGION_MASK

Read/Write Mask for MPU_RBAR, MPU_RLAR, SAU_RBAR, SAU_RLAR. Bits[4:0] of this parameter are ignored

Type

int

Default value

0xffffffff

MPU_TYPE_NS.DREGION

Number of regions in the Non-Secure MPU. If Security Extensions are absent, this is the total number of MPU regions

Type

int

Default value

0x10

MPU_TYPE_S.DREGION

Number of regions in the Secure MPU. If Security Extensions are absent, this is ignored

Type

int

Default value

0x10

MVE

Set whether the model has MVE support. If FPU = 0: 0=MVE not included, 1=Integer subset of MVE included. If FPU = 1: 0=MVE not included, 1=Integer subset of MVE included, 2=Integer and half and single precision floating point MVE included

Type

int

Default value

0x2

MVFR0.Double-precision

Support 8-byte floats

Type

bool

Default value

0x1

MVFR1.FP16

FP extension implements half-precision floating-point operations. 0 = Not supported; 1 = Supported.

Type

bool

Default value

0x1

MVFR1.FPHP

FP extension implements half-precision floating-point conversion instructions. 0x1: Half-precision to single-precision, 0x2: As for 0x1 and also half-precision to double-precision

Type

int

Default value

0x2

MVFR1.MVE

DEPRECATED: Use parameter MVE instead

Type

int

Default value

0x2

NUM_IRQ

Number of user interrupts

Type

int

Default value

0x10

NVIC_ITNS0

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS1**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS10**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS11**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value

NVIC_ITNS12

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS13**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS14**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS15**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS2**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS3**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS4**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS5**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS6**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value

NVIC_ITNS7

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS8**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**NVIC_ITNS9**

Each character fixes a security state target for a given external interrupt. 'N' implies NS; 'S' implies S; anything else is ignored. The largest bit is first, e.g. 'S-NN' sets external interrupts 0 and 1 to non-secure, 2 remains settable via the NVIC_ITNS register and 3 always targets secure

Type

string

Default value**REGISTER_POP_ORDER**

Order in which the registers are popped off the stack during exception return. A comma separated list of register names and ranges.

Type

string

Default value

R4-R11,R0-R3,R12,R14,RETURN_ADDR,CPSR,S0-S15,FPSCR,PADDING,S16-S31

REGISTER_PUSH_ORDER

Order in which the registers are pushed on to the stack during exception handling. A comma separated list of register names and ranges.

Type

string

Default value

R0-R3,R12,R14,RETURN_ADDR,CPSR,S0-S15,FPSCR,PADDING,S16-S31

SAU_CTRL.ALLNS

At reset, the SAU treats entire memory space as NS when the SAU is disabled if this is set

Type

bool

Default value

0x0

SAU_CTRL.ENABLE

Enable SAU at reset

Type

bool

Default value

0x0

SAU_REGION0.BADDR

Base address of SAU region0 at reset

Type

int

Default value

0x0

SAU_REGION0.ENABLE

Enable SAU region0 at reset

Type

bool

Default value

0x0

SAU_REGION0.LADDR

Limit address of SAU region0 at reset

Type

int

Default value

0x0

SAU_REGION0.NSC

Set NSC for SAU region0 at reset

Type

bool

Default value

0x0

SAU_REGION1.BADDR

Base address of SAU region1 at reset

Type

int

Default value

0x0

SAU_REGION1.ENABLE

Enable SAU region1 at reset

Type

bool

Default value

0x0

SAU_REGION1.LADDR

Limit address of SAU region1 at reset

Type

int

Default value

0x0

SAU_REGION1.NSC

Set NSC for SAU region1 at reset

Type

bool

Default value

0x0

SAU_REGION10.BADDR

Base address of SAU region10 at reset

Type

int

Default value

0x0

SAU_REGION10.ENABLE

Enable SAU region10 at reset

Type

bool

Default value

0x0

SAU_REGION10.LADDR

Limit address of SAU region10 at reset

Type

int

Default value

0x0

SAU_REGION10.NSC

Set NSC for SAU region10 at reset

Type

bool

Default value

0x0

SAU_REGION100.BADDR

Base address of SAU region100 at reset

Type

int

Default value

0x0

SAU_REGION100.ENABLE

Enable SAU region100 at reset

Type

bool

Default value

0x0

SAU_REGION100.LADDR

Limit address of SAU region100 at reset

Type

int

Default value

0x0

SAU_REGION100.NSC

Set NSC for SAU region100 at reset

Type

bool

Default value

0x0

SAU_REGION101.BADDR

Base address of SAU region101 at reset

Type

int

Default value

0x0

SAU_REGION101.ENABLE

Enable SAU region101 at reset

Type

bool

Default value

0x0

SAU_REGION101.LADDR

Limit address of SAU region101 at reset

Type

int

Default value

0x0

SAU_REGION101.NSC

Set NSC for SAU region101 at reset

Type

bool

Default value

0x0

SAU_REGION102.BADDR

Base address of SAU region102 at reset

Type

int

Default value

0x0

SAU_REGION102.ENABLE

Enable SAU region102 at reset

Type

bool

Default value

0x0

SAU_REGION102.LADDR

Limit address of SAU region102 at reset

Type

int

Default value

0x0

SAU_REGION102.NSC

Set NSC for SAU region102 at reset

Type

bool

Default value

0x0

SAU_REGION103.BADDR

Base address of SAU region103 at reset

Type

int

Default value

0x0

SAU_REGION103.ENABLE

Enable SAU region103 at reset

Type

bool

Default value

0x0

SAU_REGION103.LADDR

Limit address of SAU region103 at reset

Type

int

Default value

0x0

SAU_REGION103.NSC

Set NSC for SAU region103 at reset

Type

bool

Default value

0x0

SAU_REGION104.BADDR

Base address of SAU region104 at reset

Type

int

Default value

0x0

SAU_REGION104.ENABLE

Enable SAU region104 at reset

Type

bool

Default value

0x0

SAU_REGION104.LADDR

Limit address of SAU region104 at reset

Type

int

Default value

0x0

SAU_REGION104.NSC

Set NSC for SAU region104 at reset

Type

bool

Default value

0x0

SAU_REGION105.BADDR

Base address of SAU region105 at reset

Type

int

Default value

0x0

SAU_REGION105.ENABLE

Enable SAU region105 at reset

Type

bool

Default value

0x0

SAU_REGION105.LADDR

Limit address of SAU region105 at reset

Type

int

Default value

0x0

SAU_REGION105.NSC

Set NSC for SAU region105 at reset

Type

bool

Default value

0x0

SAU_REGION106.BADDR

Base address of SAU region106 at reset

Type

int

Default value

0x0

SAU_REGION106.ENABLE

Enable SAU region106 at reset

Type

bool

Default value

0x0

SAU_REGION106.LADDR

Limit address of SAU region106 at reset

Type

int

Default value

0x0

SAU_REGION106.NSC

Set NSC for SAU region106 at reset

Type

bool

Default value

0x0

SAU_REGION107.BADDR

Base address of SAU region107 at reset

Type

int

Default value

0x0

SAU_REGION107.ENABLE

Enable SAU region107 at reset

Type

bool

Default value

0x0

SAU_REGION107.LADDR

Limit address of SAU region107 at reset

Type

int

Default value

0x0

SAU_REGION107.NSC

Set NSC for SAU region107 at reset

Type

bool

Default value

0x0

SAU_REGION108.BADDR

Base address of SAU region108 at reset

Type

int

Default value

0x0

SAU_REGION108.ENABLE

Enable SAU region108 at reset

Type

bool

Default value

0x0

SAU_REGION108.LADDR

Limit address of SAU region108 at reset

Type

int

Default value

0x0

SAU_REGION108.NSC

Set NSC for SAU region108 at reset

Type

bool

Default value

0x0

SAU_REGION109.BADDR

Base address of SAU region109 at reset

Type

int

Default value

0x0

SAU_REGION109.ENABLE

Enable SAU region109 at reset

Type

bool

Default value

0x0

SAU_REGION109.LADDR

Limit address of SAU region109 at reset

Type

int

Default value

0x0

SAU_REGION109.NSC

Set NSC for SAU region109 at reset

Type

bool

Default value

0x0

SAU_REGION11.BADDR

Base address of SAU region11 at reset

Type

int

Default value

0x0

SAU_REGION11.ENABLE

Enable SAU region11 at reset

Type

bool

Default value

0x0

SAU_REGION11.LADDR

Limit address of SAU region11 at reset

Type

int

Default value

0x0

SAU_REGION11.NSC

Set NSC for SAU region11 at reset

Type

bool

Default value

0x0

SAU_REGION110.BADDR

Base address of SAU region110 at reset

Type

int

Default value

0x0

SAU_REGION110.ENABLE

Enable SAU region110 at reset

Type

bool

Default value

0x0

SAU_REGION110.LADDR

Limit address of SAU region110 at reset

Type

int

Default value

0x0

SAU_REGION110.NSC

Set NSC for SAU region110 at reset

Type

bool

Default value

0x0

SAU_REGION111.BADDR

Base address of SAU region111 at reset

Type

int

Default value

0x0

SAU_REGION111.ENABLE

Enable SAU region111 at reset

Type

bool

Default value

0x0

SAU_REGION111.LADDR

Limit address of SAU region111 at reset

Type

int

Default value

0x0

SAU_REGION111.NSC

Set NSC for SAU region111 at reset

Type

bool

Default value

0x0

SAU_REGION112.BADDR

Base address of SAU region112 at reset

Type

int

Default value

0x0

SAU_REGION112.ENABLE

Enable SAU region112 at reset

Type

bool

Default value

0x0

SAU_REGION112.LADDR

Limit address of SAU region112 at reset

Type

int

Default value

0x0

SAU_REGION112.NSC

Set NSC for SAU region112 at reset

Type

bool

Default value

0x0

SAU_REGION113.BADDR

Base address of SAU region113 at reset

Type

int

Default value

0x0

SAU_REGION113.ENABLE

Enable SAU region113 at reset

Type

bool

Default value

0x0

SAU_REGION113.LADDR

Limit address of SAU region113 at reset

Type

int

Default value

0x0

SAU_REGION113.NSC

Set NSC for SAU region113 at reset

Type

bool

Default value

0x0

SAU_REGION114.BADDR

Base address of SAU region114 at reset

Type

int

Default value

0x0

SAU_REGION114.ENABLE

Enable SAU region114 at reset

Type

bool

Default value

0x0

SAU_REGION114.LADDR

Limit address of SAU region114 at reset

Type

int

Default value

0x0

SAU_REGION114.NSC

Set NSC for SAU region114 at reset

Type

bool

Default value

0x0

SAU_REGION115.BADDR

Base address of SAU region115 at reset

Type

int

Default value

0x0

SAU_REGION115.ENABLE

Enable SAU region115 at reset

Type

bool

Default value

0x0

SAU_REGION115.LADDR

Limit address of SAU region115 at reset

Type

int

Default value

0x0

SAU_REGION115.NSC

Set NSC for SAU region115 at reset

Type

bool

Default value

0x0

SAU_REGION116.BADDR

Base address of SAU region116 at reset

Type

int

Default value

0x0

SAU_REGION116.ENABLE

Enable SAU region116 at reset

Type

bool

Default value

0x0

SAU_REGION116.LADDR

Limit address of SAU region116 at reset

Type

int

Default value

0x0

SAU_REGION116.NSC

Set NSC for SAU region116 at reset

Type

bool

Default value

0x0

SAU_REGION117.BADDR

Base address of SAU region117 at reset

Type

int

Default value

0x0

SAU_REGION117.ENABLE

Enable SAU region117 at reset

Type

bool

Default value

0x0

SAU_REGION117.LADDR

Limit address of SAU region117 at reset

Type

int

Default value

0x0

SAU_REGION117.NSC

Set NSC for SAU region117 at reset

Type

bool

Default value

0x0

SAU_REGION118.BADDR

Base address of SAU region118 at reset

Type

int

Default value

0x0

SAU_REGION118.ENABLE

Enable SAU region118 at reset

Type

bool

Default value

0x0

SAU_REGION118.LADDR

Limit address of SAU region118 at reset

Type

int

Default value

0x0

SAU_REGION118.NSC

Set NSC for SAU region118 at reset

Type

bool

Default value

0x0

SAU_REGION119.BADDR

Base address of SAU region119 at reset

Type

int

Default value

0x0

SAU_REGION119.ENABLE

Enable SAU region119 at reset

Type

bool

Default value

0x0

SAU_REGION119.LADDR

Limit address of SAU region119 at reset

Type

int

Default value

0x0

SAU_REGION119.NSC

Set NSC for SAU region119 at reset

Type

bool

Default value

0x0

SAU_REGION12.BADDR

Base address of SAU region12 at reset

Type

int

Default value

0x0

SAU_REGION12.ENABLE

Enable SAU region12 at reset

Type

bool

Default value

0x0

SAU_REGION12.LADDR

Limit address of SAU region12 at reset

Type

int

Default value

0x0

SAU_REGION12.NSC

Set NSC for SAU region12 at reset

Type

bool

Default value

0x0

SAU_REGION120.BADDR

Base address of SAU region120 at reset

Type

int

Default value

0x0

SAU_REGION120.ENABLE

Enable SAU region120 at reset

Type

bool

Default value

0x0

SAU_REGION120.LADDR

Limit address of SAU region120 at reset

Type

int

Default value

0x0

SAU_REGION120.NSC

Set NSC for SAU region120 at reset

Type

bool

Default value

0x0

SAU_REGION121.BADDR

Base address of SAU region121 at reset

Type

int

Default value

0x0

SAU_REGION121.ENABLE

Enable SAU region121 at reset

Type

bool

Default value

0x0

SAU_REGION121.LADDR

Limit address of SAU region121 at reset

Type

int

Default value

0x0

SAU_REGION121.NSC

Set NSC for SAU region121 at reset

Type

bool

Default value

0x0

SAU_REGION122.BADDR

Base address of SAU region122 at reset

Type

int

Default value

0x0

SAU_REGION122.ENABLE

Enable SAU region122 at reset

Type

bool

Default value

0x0

SAU_REGION122.LADDR

Limit address of SAU region122 at reset

Type

int

Default value

0x0

SAU_REGION122.NSC

Set NSC for SAU region122 at reset

Type

bool

Default value

0x0

SAU_REGION123.BADDR

Base address of SAU region123 at reset

Type

int

Default value

0x0

SAU_REGION123.ENABLE

Enable SAU region123 at reset

Type

bool

Default value

0x0

SAU_REGION123.LADDR

Limit address of SAU region123 at reset

Type

int

Default value

0x0

SAU_REGION123.NSC

Set NSC for SAU region123 at reset

Type

bool

Default value

0x0

SAU_REGION124.BADDR

Base address of SAU region124 at reset

Type

int

Default value

0x0

SAU_REGION124.ENABLE

Enable SAU region124 at reset

Type

bool

Default value

0x0

SAU_REGION124.LADDR

Limit address of SAU region124 at reset

Type

int

Default value

0x0

SAU_REGION124.NSC

Set NSC for SAU region124 at reset

Type

bool

Default value

0x0

SAU_REGION125.BADDR

Base address of SAU region125 at reset

Type

int

Default value

0x0

SAU_REGION125.ENABLE

Enable SAU region125 at reset

Type

bool

Default value

0x0

SAU_REGION125.LADDR

Limit address of SAU region125 at reset

Type

int

Default value

0x0

SAU_REGION125.NSC

Set NSC for SAU region125 at reset

Type

bool

Default value

0x0

SAU_REGION126.BADDR

Base address of SAU region126 at reset

Type

int

Default value

0x0

SAU_REGION126.ENABLE

Enable SAU region126 at reset

Type

bool

Default value

0x0

SAU_REGION126.LADDR

Limit address of SAU region126 at reset

Type

int

Default value

0x0

SAU_REGION126.NSC

Set NSC for SAU region126 at reset

Type

bool

Default value

0x0

SAU_REGION127.BADDR

Base address of SAU region127 at reset

Type

int

Default value

0x0

SAU_REGION127.ENABLE

Enable SAU region127 at reset

Type

bool

Default value

0x0

SAU_REGION127.LADDR

Limit address of SAU region127 at reset

Type

int

Default value

0x0

SAU_REGION127.NSC

Set NSC for SAU region127 at reset

Type

bool

Default value

0x0

SAU_REGION128.BADDR

Base address of SAU region128 at reset

Type

int

Default value

0x0

SAU_REGION128.ENABLE

Enable SAU region128 at reset

Type

bool

Default value

0x0

SAU_REGION128.LADDR

Limit address of SAU region128 at reset

Type

int

Default value

0x0

SAU_REGION128.NSC

Set NSC for SAU region128 at reset

Type

bool

Default value

0x0

SAU_REGION129.BADDR

Base address of SAU region129 at reset

Type

int

Default value

0x0

SAU_REGION129.ENABLE

Enable SAU region129 at reset

Type

bool

Default value

0x0

SAU_REGION129.LADDR

Limit address of SAU region129 at reset

Type

int

Default value

0x0

SAU_REGION129.NSC

Set NSC for SAU region129 at reset

Type

bool

Default value

0x0

SAU_REGION13.BADDR

Base address of SAU region13 at reset

Type

int

Default value

0x0

SAU_REGION13.ENABLE

Enable SAU region13 at reset

Type

bool

Default value

0x0

SAU_REGION13.LADDR

Limit address of SAU region13 at reset

Type

int

Default value

0x0

SAU_REGION13.NSC

Set NSC for SAU region13 at reset

Type

bool

Default value

0x0

SAU_REGION130.BADDR

Base address of SAU region130 at reset

Type

int

Default value

0x0

SAU_REGION130.ENABLE

Enable SAU region130 at reset

Type

bool

Default value

0x0

SAU_REGION130.LADDR

Limit address of SAU region130 at reset

Type

int

Default value

0x0

SAU_REGION130.NSC

Set NSC for SAU region130 at reset

Type

bool

Default value

0x0

SAU_REGION131.BADDR

Base address of SAU region131 at reset

Type

int

Default value

0x0

SAU_REGION131.ENABLE

Enable SAU region131 at reset

Type

bool

Default value

0x0

SAU_REGION131.LADDR

Limit address of SAU region131 at reset

Type

int

Default value

0x0

SAU_REGION131.NSC

Set NSC for SAU region131 at reset

Type

bool

Default value

0x0

SAU_REGION132.BADDR

Base address of SAU region132 at reset

Type

int

Default value

0x0

SAU_REGION132.ENABLE

Enable SAU region132 at reset

Type

bool

Default value

0x0

SAU_REGION132.LADDR

Limit address of SAU region132 at reset

Type

int

Default value

0x0

SAU_REGION132.NSC

Set NSC for SAU region132 at reset

Type

bool

Default value

0x0

SAU_REGION133.BADDR

Base address of SAU region133 at reset

Type

int

Default value

0x0

SAU_REGION133.ENABLE

Enable SAU region133 at reset

Type

bool

Default value

0x0

SAU_REGION133.LADDR

Limit address of SAU region133 at reset

Type

int

Default value

0x0

SAU_REGION133.NSC

Set NSC for SAU region133 at reset

Type

bool

Default value

0x0

SAU_REGION134.BADDR

Base address of SAU region134 at reset

Type

int

Default value

0x0

SAU_REGION134.ENABLE

Enable SAU region134 at reset

Type

bool

Default value

0x0

SAU_REGION134.LADDR

Limit address of SAU region134 at reset

Type

int

Default value

0x0

SAU_REGION134.NSC

Set NSC for SAU region134 at reset

Type

bool

Default value

0x0

SAU_REGION135.BADDR

Base address of SAU region135 at reset

Type

int

Default value

0x0

SAU_REGION135.ENABLE

Enable SAU region135 at reset

Type

bool

Default value

0x0

SAU_REGION135.LADDR

Limit address of SAU region135 at reset

Type

int

Default value

0x0

SAU_REGION135.NSC

Set NSC for SAU region135 at reset

Type

bool

Default value

0x0

SAU_REGION136.BADDR

Base address of SAU region136 at reset

Type

int

Default value

0x0

SAU_REGION136.ENABLE

Enable SAU region136 at reset

Type

bool

Default value

0x0

SAU_REGION136.LADDR

Limit address of SAU region136 at reset

Type

int

Default value

0x0

SAU_REGION136.NSC

Set NSC for SAU region136 at reset

Type

bool

Default value

0x0

SAU_REGION137.BADDR

Base address of SAU region137 at reset

Type

int

Default value

0x0

SAU_REGION137.ENABLE

Enable SAU region137 at reset

Type

bool

Default value

0x0

SAU_REGION137.LADDR

Limit address of SAU region137 at reset

Type

int

Default value

0x0

SAU_REGION137.NSC

Set NSC for SAU region137 at reset

Type

bool

Default value

0x0

SAU_REGION138.BADDR

Base address of SAU region138 at reset

Type

int

Default value

0x0

SAU_REGION138.ENABLE

Enable SAU region138 at reset

Type

bool

Default value

0x0

SAU_REGION138.LADDR

Limit address of SAU region138 at reset

Type

int

Default value

0x0

SAU_REGION138.NSC

Set NSC for SAU region138 at reset

Type

bool

Default value

0x0

SAU_REGION139.BADDR

Base address of SAU region139 at reset

Type

int

Default value

0x0

SAU_REGION139.ENABLE

Enable SAU region139 at reset

Type

bool

Default value

0x0

SAU_REGION139.LADDR

Limit address of SAU region139 at reset

Type

int

Default value

0x0

SAU_REGION139.NSC

Set NSC for SAU region139 at reset

Type

bool

Default value

0x0

SAU_REGION14.BADDR

Base address of SAU region14 at reset

Type

int

Default value

0x0

SAU_REGION14.ENABLE

Enable SAU region14 at reset

Type

bool

Default value

0x0

SAU_REGION14.LADDR

Limit address of SAU region14 at reset

Type

int

Default value

0x0

SAU_REGION14.NSC

Set NSC for SAU region14 at reset

Type

bool

Default value

0x0

SAU_REGION140.BADDR

Base address of SAU region140 at reset

Type

int

Default value

0x0

SAU_REGION140.ENABLE

Enable SAU region140 at reset

Type

bool

Default value

0x0

SAU_REGION140.LADDR

Limit address of SAU region140 at reset

Type

int

Default value

0x0

SAU_REGION140.NSC

Set NSC for SAU region140 at reset

Type

bool

Default value

0x0

SAU_REGION141.BADDR

Base address of SAU region141 at reset

Type

int

Default value

0x0

SAU_REGION141.ENABLE

Enable SAU region141 at reset

Type

bool

Default value

0x0

SAU_REGION141.LADDR

Limit address of SAU region141 at reset

Type

int

Default value

0x0

SAU_REGION141.NSC

Set NSC for SAU region141 at reset

Type

bool

Default value

0x0

SAU_REGION142.BADDR

Base address of SAU region142 at reset

Type

int

Default value

0x0

SAU_REGION142.ENABLE

Enable SAU region142 at reset

Type

bool

Default value

0x0

SAU_REGION142.LADDR

Limit address of SAU region142 at reset

Type

int

Default value

0x0

SAU_REGION142.NSC

Set NSC for SAU region142 at reset

Type

bool

Default value

0x0

SAU_REGION143.BADDR

Base address of SAU region143 at reset

Type

int

Default value

0x0

SAU_REGION143.ENABLE

Enable SAU region143 at reset

Type

bool

Default value

0x0

SAU_REGION143.LADDR

Limit address of SAU region143 at reset

Type

int

Default value

0x0

SAU_REGION143.NSC

Set NSC for SAU region143 at reset

Type

bool

Default value

0x0

SAU_REGION144.BADDR

Base address of SAU region144 at reset

Type

int

Default value

0x0

SAU_REGION144.ENABLE

Enable SAU region144 at reset

Type

bool

Default value

0x0

SAU_REGION144.LADDR

Limit address of SAU region144 at reset

Type

int

Default value

0x0

SAU_REGION144.NSC

Set NSC for SAU region144 at reset

Type

bool

Default value

0x0

SAU_REGION145.BADDR

Base address of SAU region145 at reset

Type

int

Default value

0x0

SAU_REGION145.ENABLE

Enable SAU region145 at reset

Type

bool

Default value

0x0

SAU_REGION145.LADDR

Limit address of SAU region145 at reset

Type

int

Default value

0x0

SAU_REGION145.NSC

Set NSC for SAU region145 at reset

Type

bool

Default value

0x0

SAU_REGION146.BADDR

Base address of SAU region146 at reset

Type

int

Default value

0x0

SAU_REGION146.ENABLE

Enable SAU region146 at reset

Type

bool

Default value

0x0

SAU_REGION146.LADDR

Limit address of SAU region146 at reset

Type

int

Default value

0x0

SAU_REGION146.NSC

Set NSC for SAU region146 at reset

Type

bool

Default value

0x0

SAU_REGION147.BADDR

Base address of SAU region147 at reset

Type

int

Default value

0x0

SAU_REGION147.ENABLE

Enable SAU region147 at reset

Type

bool

Default value

0x0

SAU_REGION147.LADDR

Limit address of SAU region147 at reset

Type

int

Default value

0x0

SAU_REGION147.NSC

Set NSC for SAU region147 at reset

Type

bool

Default value

0x0

SAU_REGION148.BADDR

Base address of SAU region148 at reset

Type

int

Default value

0x0

SAU_REGION148.ENABLE

Enable SAU region148 at reset

Type

bool

Default value

0x0

SAU_REGION148.LADDR

Limit address of SAU region148 at reset

Type

int

Default value

0x0

SAU_REGION148.NSC

Set NSC for SAU region148 at reset

Type

bool

Default value

0x0

SAU_REGION149.BADDR

Base address of SAU region149 at reset

Type

int

Default value

0x0

SAU_REGION149.ENABLE

Enable SAU region149 at reset

Type

bool

Default value

0x0

SAU_REGION149.LADDR

Limit address of SAU region149 at reset

Type

int

Default value

0x0

SAU_REGION149.NSC

Set NSC for SAU region149 at reset

Type

bool

Default value

0x0

SAU_REGION15.BADDR

Base address of SAU region15 at reset

Type

int

Default value

0x0

SAU_REGION15.ENABLE

Enable SAU region15 at reset

Type

bool

Default value

0x0

SAU_REGION15.LADDR

Limit address of SAU region15 at reset

Type

int

Default value

0x0

SAU_REGION15.NSC

Set NSC for SAU region15 at reset

Type

bool

Default value

0x0

SAU_REGION150.BADDR

Base address of SAU region150 at reset

Type

int

Default value

0x0

SAU_REGION150.ENABLE

Enable SAU region150 at reset

Type

bool

Default value

0x0

SAU_REGION150.LADDR

Limit address of SAU region150 at reset

Type

int

Default value

0x0

SAU_REGION150.NSC

Set NSC for SAU region150 at reset

Type

bool

Default value

0x0

SAU_REGION151.BADDR

Base address of SAU region151 at reset

Type

int

Default value

0x0

SAU_REGION151.ENABLE

Enable SAU region151 at reset

Type

bool

Default value

0x0

SAU_REGION151.LADDR

Limit address of SAU region151 at reset

Type

int

Default value

0x0

SAU_REGION151.NSC

Set NSC for SAU region151 at reset

Type

bool

Default value

0x0

SAU_REGION152.BADDR

Base address of SAU region152 at reset

Type

int

Default value

0x0

SAU_REGION152.ENABLE

Enable SAU region152 at reset

Type

bool

Default value

0x0

SAU_REGION152.LADDR

Limit address of SAU region152 at reset

Type

int

Default value

0x0

SAU_REGION152.NSC

Set NSC for SAU region152 at reset

Type

bool

Default value

0x0

SAU_REGION153.BADDR

Base address of SAU region153 at reset

Type

int

Default value

0x0

SAU_REGION153.ENABLE

Enable SAU region153 at reset

Type

bool

Default value

0x0

SAU_REGION153.LADDR

Limit address of SAU region153 at reset

Type

int

Default value

0x0

SAU_REGION153.NSC

Set NSC for SAU region153 at reset

Type

bool

Default value

0x0

SAU_REGION154.BADDR

Base address of SAU region154 at reset

Type

int

Default value

0x0

SAU_REGION154.ENABLE

Enable SAU region154 at reset

Type

bool

Default value

0x0

SAU_REGION154.LADDR

Limit address of SAU region154 at reset

Type

int

Default value

0x0

SAU_REGION154.NSC

Set NSC for SAU region154 at reset

Type

bool

Default value

0x0

SAU_REGION155.BADDR

Base address of SAU region155 at reset

Type

int

Default value

0x0

SAU_REGION155.ENABLE

Enable SAU region155 at reset

Type

bool

Default value

0x0

SAU_REGION155.LADDR

Limit address of SAU region155 at reset

Type

int

Default value

0x0

SAU_REGION155.NSC

Set NSC for SAU region155 at reset

Type

bool

Default value

0x0

SAU_REGION156.BADDR

Base address of SAU region156 at reset

Type

int

Default value

0x0

SAU_REGION156.ENABLE

Enable SAU region156 at reset

Type

bool

Default value

0x0

SAU_REGION156.LADDR

Limit address of SAU region156 at reset

Type

int

Default value

0x0

SAU_REGION156.NSC

Set NSC for SAU region156 at reset

Type

bool

Default value

0x0

SAU_REGION157.BADDR

Base address of SAU region157 at reset

Type

int

Default value

0x0

SAU_REGION157.ENABLE

Enable SAU region157 at reset

Type

bool

Default value

0x0

SAU_REGION157.LADDR

Limit address of SAU region157 at reset

Type

int

Default value

0x0

SAU_REGION157.NSC

Set NSC for SAU region157 at reset

Type

bool

Default value

0x0

SAU_REGION158.BADDR

Base address of SAU region158 at reset

Type

int

Default value

0x0

SAU_REGION158.ENABLE

Enable SAU region158 at reset

Type

bool

Default value

0x0

SAU_REGION158.LADDR

Limit address of SAU region158 at reset

Type

int

Default value

0x0

SAU_REGION158.NSC

Set NSC for SAU region158 at reset

Type

bool

Default value

0x0

SAU_REGION159.BADDR

Base address of SAU region159 at reset

Type

int

Default value

0x0

SAU_REGION159.ENABLE

Enable SAU region159 at reset

Type

bool

Default value

0x0

SAU_REGION159.LADDR

Limit address of SAU region159 at reset

Type

int

Default value

0x0

SAU_REGION159.NSC

Set NSC for SAU region159 at reset

Type

bool

Default value

0x0

SAU_REGION16.BADDR

Base address of SAU region16 at reset

Type

int

Default value

0x0

SAU_REGION16.ENABLE

Enable SAU region16 at reset

Type

bool

Default value

0x0

SAU_REGION16.LADDR

Limit address of SAU region16 at reset

Type

int

Default value

0x0

SAU_REGION16.NSC

Set NSC for SAU region16 at reset

Type

bool

Default value

0x0

SAU_REGION160.BADDR

Base address of SAU region160 at reset

Type

int

Default value

0x0

SAU_REGION160.ENABLE

Enable SAU region160 at reset

Type

bool

Default value

0x0

SAU_REGION160.LADDR

Limit address of SAU region160 at reset

Type

int

Default value

0x0

SAU_REGION160.NSC

Set NSC for SAU region160 at reset

Type

bool

Default value

0x0

SAU_REGION161.BADDR

Base address of SAU region161 at reset

Type

int

Default value

0x0

SAU_REGION161.ENABLE

Enable SAU region161 at reset

Type

bool

Default value

0x0

SAU_REGION161.LADDR

Limit address of SAU region161 at reset

Type

int

Default value

0x0

SAU_REGION161.NSC

Set NSC for SAU region161 at reset

Type

bool

Default value

0x0

SAU_REGION162.BADDR

Base address of SAU region162 at reset

Type

int

Default value

0x0

SAU_REGION162.ENABLE

Enable SAU region162 at reset

Type

bool

Default value

0x0

SAU_REGION162.LADDR

Limit address of SAU region162 at reset

Type

int

Default value

0x0

SAU_REGION162.NSC

Set NSC for SAU region162 at reset

Type

bool

Default value

0x0

SAU_REGION163.BADDR

Base address of SAU region163 at reset

Type

int

Default value

0x0

SAU_REGION163.ENABLE

Enable SAU region163 at reset

Type

bool

Default value

0x0

SAU_REGION163.LADDR

Limit address of SAU region163 at reset

Type

int

Default value

0x0

SAU_REGION163.NSC

Set NSC for SAU region163 at reset

Type

bool

Default value

0x0

SAU_REGION164.BADDR

Base address of SAU region164 at reset

Type

int

Default value

0x0

SAU_REGION164.ENABLE

Enable SAU region164 at reset

Type

bool

Default value

0x0

SAU_REGION164.LADDR

Limit address of SAU region164 at reset

Type

int

Default value

0x0

SAU_REGION164.NSC

Set NSC for SAU region164 at reset

Type

bool

Default value

0x0

SAU_REGION165.BADDR

Base address of SAU region165 at reset

Type

int

Default value

0x0

SAU_REGION165.ENABLE

Enable SAU region165 at reset

Type

bool

Default value

0x0

SAU_REGION165.LADDR

Limit address of SAU region165 at reset

Type

int

Default value

0x0

SAU_REGION165.NSC

Set NSC for SAU region165 at reset

Type

bool

Default value

0x0

SAU_REGION166.BADDR

Base address of SAU region166 at reset

Type

int

Default value

0x0

SAU_REGION166.ENABLE

Enable SAU region166 at reset

Type

bool

Default value

0x0

SAU_REGION166.LADDR

Limit address of SAU region166 at reset

Type

int

Default value

0x0

SAU_REGION166.NSC

Set NSC for SAU region166 at reset

Type

bool

Default value

0x0

SAU_REGION167.BADDR

Base address of SAU region167 at reset

Type

int

Default value

0x0

SAU_REGION167.ENABLE

Enable SAU region167 at reset

Type

bool

Default value

0x0

SAU_REGION167.LADDR

Limit address of SAU region167 at reset

Type

int

Default value

0x0

SAU_REGION167.NSC

Set NSC for SAU region167 at reset

Type

bool

Default value

0x0

SAU_REGION168.BADDR

Base address of SAU region168 at reset

Type

int

Default value

0x0

SAU_REGION168.ENABLE

Enable SAU region168 at reset

Type

bool

Default value

0x0

SAU_REGION168.LADDR

Limit address of SAU region168 at reset

Type

int

Default value

0x0

SAU_REGION168.NSC

Set NSC for SAU region168 at reset

Type

bool

Default value

0x0

SAU_REGION169.BADDR

Base address of SAU region169 at reset

Type

int

Default value

0x0

SAU_REGION169.ENABLE

Enable SAU region169 at reset

Type

bool

Default value

0x0

SAU_REGION169.LADDR

Limit address of SAU region169 at reset

Type

int

Default value

0x0

SAU_REGION169.NSC

Set NSC for SAU region169 at reset

Type

bool

Default value

0x0

SAU_REGION17.BADDR

Base address of SAU region17 at reset

Type

int

Default value

0x0

SAU_REGION17.ENABLE

Enable SAU region17 at reset

Type

bool

Default value

0x0

SAU_REGION17.LADDR

Limit address of SAU region17 at reset

Type

int

Default value

0x0

SAU_REGION17.NSC

Set NSC for SAU region17 at reset

Type

bool

Default value

0x0

SAU_REGION170.BADDR

Base address of SAU region170 at reset

Type

int

Default value

0x0

SAU_REGION170.ENABLE

Enable SAU region170 at reset

Type

bool

Default value

0x0

SAU_REGION170.LADDR

Limit address of SAU region170 at reset

Type

int

Default value

0x0

SAU_REGION170.NSC

Set NSC for SAU region170 at reset

Type

bool

Default value

0x0

SAU_REGION171.BADDR

Base address of SAU region171 at reset

Type

int

Default value

0x0

SAU_REGION171.ENABLE

Enable SAU region171 at reset

Type

bool

Default value

0x0

SAU_REGION171.LADDR

Limit address of SAU region171 at reset

Type

int

Default value

0x0

SAU_REGION171.NSC

Set NSC for SAU region171 at reset

Type

bool

Default value

0x0

SAU_REGION172.BADDR

Base address of SAU region172 at reset

Type

int

Default value

0x0

SAU_REGION172.ENABLE

Enable SAU region172 at reset

Type

bool

Default value

0x0

SAU_REGION172.LADDR

Limit address of SAU region172 at reset

Type

int

Default value

0x0

SAU_REGION172.NSC

Set NSC for SAU region172 at reset

Type

bool

Default value

0x0

SAU_REGION173.BADDR

Base address of SAU region173 at reset

Type

int

Default value

0x0

SAU_REGION173.ENABLE

Enable SAU region173 at reset

Type

bool

Default value

0x0

SAU_REGION173.LADDR

Limit address of SAU region173 at reset

Type

int

Default value

0x0

SAU_REGION173.NSC

Set NSC for SAU region173 at reset

Type

bool

Default value

0x0

SAU_REGION174.BADDR

Base address of SAU region174 at reset

Type

int

Default value

0x0

SAU_REGION174.ENABLE

Enable SAU region174 at reset

Type

bool

Default value

0x0

SAU_REGION174.LADDR

Limit address of SAU region174 at reset

Type

int

Default value

0x0

SAU_REGION174.NSC

Set NSC for SAU region174 at reset

Type

bool

Default value

0x0

SAU_REGION175.BADDR

Base address of SAU region175 at reset

Type

int

Default value

0x0

SAU_REGION175.ENABLE

Enable SAU region175 at reset

Type

bool

Default value

0x0

SAU_REGION175.LADDR

Limit address of SAU region175 at reset

Type

int

Default value

0x0

SAU_REGION175.NSC

Set NSC for SAU region175 at reset

Type

bool

Default value

0x0

SAU_REGION176.BADDR

Base address of SAU region176 at reset

Type

int

Default value

0x0

SAU_REGION176.ENABLE

Enable SAU region176 at reset

Type

bool

Default value

0x0

SAU_REGION176.LADDR

Limit address of SAU region176 at reset

Type

int

Default value

0x0

SAU_REGION176.NSC

Set NSC for SAU region176 at reset

Type

bool

Default value

0x0

SAU_REGION177.BADDR

Base address of SAU region177 at reset

Type

int

Default value

0x0

SAU_REGION177.ENABLE

Enable SAU region177 at reset

Type

bool

Default value

0x0

SAU_REGION177.LADDR

Limit address of SAU region177 at reset

Type

int

Default value

0x0

SAU_REGION177.NSC

Set NSC for SAU region177 at reset

Type

bool

Default value

0x0

SAU_REGION178.BADDR

Base address of SAU region178 at reset

Type

int

Default value

0x0

SAU_REGION178.ENABLE

Enable SAU region178 at reset

Type

bool

Default value

0x0

SAU_REGION178.LADDR

Limit address of SAU region178 at reset

Type

int

Default value

0x0

SAU_REGION178.NSC

Set NSC for SAU region178 at reset

Type

bool

Default value

0x0

SAU_REGION179.BADDR

Base address of SAU region179 at reset

Type

int

Default value

0x0

SAU_REGION179.ENABLE

Enable SAU region179 at reset

Type

bool

Default value

0x0

SAU_REGION179.LADDR

Limit address of SAU region179 at reset

Type

int

Default value

0x0

SAU_REGION179.NSC

Set NSC for SAU region179 at reset

Type

bool

Default value

0x0

SAU_REGION18.BADDR

Base address of SAU region18 at reset

Type

int

Default value

0x0

SAU_REGION18.ENABLE

Enable SAU region18 at reset

Type

bool

Default value

0x0

SAU_REGION18.LADDR

Limit address of SAU region18 at reset

Type

int

Default value

0x0

SAU_REGION18.NSC

Set NSC for SAU region18 at reset

Type

bool

Default value

0x0

SAU_REGION180.BADDR

Base address of SAU region180 at reset

Type

int

Default value

0x0

SAU_REGION180.ENABLE

Enable SAU region180 at reset

Type

bool

Default value

0x0

SAU_REGION180.LADDR

Limit address of SAU region180 at reset

Type

int

Default value

0x0

SAU_REGION180.NSC

Set NSC for SAU region180 at reset

Type

bool

Default value

0x0

SAU_REGION181.BADDR

Base address of SAU region181 at reset

Type

int

Default value

0x0

SAU_REGION181.ENABLE

Enable SAU region181 at reset

Type

bool

Default value

0x0

SAU_REGION181.LADDR

Limit address of SAU region181 at reset

Type

int

Default value

0x0

SAU_REGION181.NSC

Set NSC for SAU region181 at reset

Type

bool

Default value

0x0

SAU_REGION182.BADDR

Base address of SAU region182 at reset

Type

int

Default value

0x0

SAU_REGION182.ENABLE

Enable SAU region182 at reset

Type

bool

Default value

0x0

SAU_REGION182.LADDR

Limit address of SAU region182 at reset

Type

int

Default value

0x0

SAU_REGION182.NSC

Set NSC for SAU region182 at reset

Type

bool

Default value

0x0

SAU_REGION183.BADDR

Base address of SAU region183 at reset

Type

int

Default value

0x0

SAU_REGION183.ENABLE

Enable SAU region183 at reset

Type

bool

Default value

0x0

SAU_REGION183.LADDR

Limit address of SAU region183 at reset

Type

int

Default value

0x0

SAU_REGION183.NSC

Set NSC for SAU region183 at reset

Type

bool

Default value

0x0

SAU_REGION184.BADDR

Base address of SAU region184 at reset

Type

int

Default value

0x0

SAU_REGION184.ENABLE

Enable SAU region184 at reset

Type

bool

Default value

0x0

SAU_REGION184.LADDR

Limit address of SAU region184 at reset

Type

int

Default value

0x0

SAU_REGION184.NSC

Set NSC for SAU region184 at reset

Type

bool

Default value

0x0

SAU_REGION185.BADDR

Base address of SAU region185 at reset

Type

int

Default value

0x0

SAU_REGION185.ENABLE

Enable SAU region185 at reset

Type

bool

Default value

0x0

SAU_REGION185.LADDR

Limit address of SAU region185 at reset

Type

int

Default value

0x0

SAU_REGION185.NSC

Set NSC for SAU region185 at reset

Type

bool

Default value

0x0

SAU_REGION186.BADDR

Base address of SAU region186 at reset

Type

int

Default value

0x0

SAU_REGION186.ENABLE

Enable SAU region186 at reset

Type

bool

Default value

0x0

SAU_REGION186.LADDR

Limit address of SAU region186 at reset

Type

int

Default value

0x0

SAU_REGION186.NSC

Set NSC for SAU region186 at reset

Type

bool

Default value

0x0

SAU_REGION187.BADDR

Base address of SAU region187 at reset

Type

int

Default value

0x0

SAU_REGION187.ENABLE

Enable SAU region187 at reset

Type

bool

Default value

0x0

SAU_REGION187.LADDR

Limit address of SAU region187 at reset

Type

int

Default value

0x0

SAU_REGION187.NSC

Set NSC for SAU region187 at reset

Type

bool

Default value

0x0

SAU_REGION188.BADDR

Base address of SAU region188 at reset

Type

int

Default value

0x0

SAU_REGION188.ENABLE

Enable SAU region188 at reset

Type

bool

Default value

0x0

SAU_REGION188.LADDR

Limit address of SAU region188 at reset

Type

int

Default value

0x0

SAU_REGION188.NSC

Set NSC for SAU region188 at reset

Type

bool

Default value

0x0

SAU_REGION189.BADDR

Base address of SAU region189 at reset

Type

int

Default value

0x0

SAU_REGION189.ENABLE

Enable SAU region189 at reset

Type

bool

Default value

0x0

SAU_REGION189.LADDR

Limit address of SAU region189 at reset

Type

int

Default value

0x0

SAU_REGION189.NSC

Set NSC for SAU region189 at reset

Type

bool

Default value

0x0

SAU_REGION19.BADDR

Base address of SAU region19 at reset

Type

int

Default value

0x0

SAU_REGION19.ENABLE

Enable SAU region19 at reset

Type

bool

Default value

0x0

SAU_REGION19.LADDR

Limit address of SAU region19 at reset

Type

int

Default value

0x0

SAU_REGION19.NSC

Set NSC for SAU region19 at reset

Type

bool

Default value

0x0

SAU_REGION190.BADDR

Base address of SAU region190 at reset

Type

int

Default value

0x0

SAU_REGION190.ENABLE

Enable SAU region190 at reset

Type

bool

Default value

0x0

SAU_REGION190.LADDR

Limit address of SAU region190 at reset

Type

int

Default value

0x0

SAU_REGION190.NSC

Set NSC for SAU region190 at reset

Type

bool

Default value

0x0

SAU_REGION191.BADDR

Base address of SAU region191 at reset

Type

int

Default value

0x0

SAU_REGION191.ENABLE

Enable SAU region191 at reset

Type

bool

Default value

0x0

SAU_REGION191.LADDR

Limit address of SAU region191 at reset

Type

int

Default value

0x0

SAU_REGION191.NSC

Set NSC for SAU region191 at reset

Type

bool

Default value

0x0

SAU_REGION192.BADDR

Base address of SAU region192 at reset

Type

int

Default value

0x0

SAU_REGION192.ENABLE

Enable SAU region192 at reset

Type

bool

Default value

0x0

SAU_REGION192.LADDR

Limit address of SAU region192 at reset

Type

int

Default value

0x0

SAU_REGION192.NSC

Set NSC for SAU region192 at reset

Type

bool

Default value

0x0

SAU_REGION193.BADDR

Base address of SAU region193 at reset

Type

int

Default value

0x0

SAU_REGION193.ENABLE

Enable SAU region193 at reset

Type

bool

Default value

0x0

SAU_REGION193.LADDR

Limit address of SAU region193 at reset

Type

int

Default value

0x0

SAU_REGION193.NSC

Set NSC for SAU region193 at reset

Type

bool

Default value

0x0

SAU_REGION194.BADDR

Base address of SAU region194 at reset

Type

int

Default value

0x0

SAU_REGION194.ENABLE

Enable SAU region194 at reset

Type

bool

Default value

0x0

SAU_REGION194.LADDR

Limit address of SAU region194 at reset

Type

int

Default value

0x0

SAU_REGION194.NSC

Set NSC for SAU region194 at reset

Type

bool

Default value

0x0

SAU_REGION195.BADDR

Base address of SAU region195 at reset

Type

int

Default value

0x0

SAU_REGION195.ENABLE

Enable SAU region195 at reset

Type

bool

Default value

0x0

SAU_REGION195.LADDR

Limit address of SAU region195 at reset

Type

int

Default value

0x0

SAU_REGION195.NSC

Set NSC for SAU region195 at reset

Type

bool

Default value

0x0

SAU_REGION196.BADDR

Base address of SAU region196 at reset

Type

int

Default value

0x0

SAU_REGION196.ENABLE

Enable SAU region196 at reset

Type

bool

Default value

0x0

SAU_REGION196.LADDR

Limit address of SAU region196 at reset

Type

int

Default value

0x0

SAU_REGION196.NSC

Set NSC for SAU region196 at reset

Type

bool

Default value

0x0

SAU_REGION197.BADDR

Base address of SAU region197 at reset

Type

int

Default value

0x0

SAU_REGION197.ENABLE

Enable SAU region197 at reset

Type

bool

Default value

0x0

SAU_REGION197.LADDR

Limit address of SAU region197 at reset

Type

int

Default value

0x0

SAU_REGION197.NSC

Set NSC for SAU region197 at reset

Type

bool

Default value

0x0

SAU_REGION198.BADDR

Base address of SAU region198 at reset

Type

int

Default value

0x0

SAU_REGION198.ENABLE

Enable SAU region198 at reset

Type

bool

Default value

0x0

SAU_REGION198.LADDR

Limit address of SAU region198 at reset

Type

int

Default value

0x0

SAU_REGION198.NSC

Set NSC for SAU region198 at reset

Type

bool

Default value

0x0

SAU_REGION199.BADDR

Base address of SAU region199 at reset

Type

int

Default value

0x0

SAU_REGION199.ENABLE

Enable SAU region199 at reset

Type

bool

Default value

0x0

SAU_REGION199.LADDR

Limit address of SAU region199 at reset

Type

int

Default value

0x0

SAU_REGION199.NSC

Set NSC for SAU region199 at reset

Type

bool

Default value

0x0

SAU_REGION2.BADDR

Base address of SAU region2 at reset

Type

int

Default value

0x0

SAU_REGION2.ENABLE

Enable SAU region2 at reset

Type

bool

Default value

0x0

SAU_REGION2.LADDR

Limit address of SAU region2 at reset

Type

int

Default value

0x0

SAU_REGION2.NSC

Set NSC for SAU region2 at reset

Type

bool

Default value

0x0

SAU_REGION20.BADDR

Base address of SAU region20 at reset

Type

int

Default value

0x0

SAU_REGION20.ENABLE

Enable SAU region20 at reset

Type

bool

Default value

0x0

SAU_REGION20.LADDR

Limit address of SAU region20 at reset

Type

int

Default value

0x0

SAU_REGION20.NSC

Set NSC for SAU region20 at reset

Type

bool

Default value

0x0

SAU_REGION200.BADDR

Base address of SAU region200 at reset

Type

int

Default value

0x0

SAU_REGION200.ENABLE

Enable SAU region200 at reset

Type

bool

Default value

0x0

SAU_REGION200.LADDR

Limit address of SAU region200 at reset

Type

int

Default value

0x0

SAU_REGION200.NSC

Set NSC for SAU region200 at reset

Type

bool

Default value

0x0

SAU_REGION201.BADDR

Base address of SAU region201 at reset

Type

int

Default value

0x0

SAU_REGION201.ENABLE

Enable SAU region201 at reset

Type

bool

Default value

0x0

SAU_REGION201.LADDR

Limit address of SAU region201 at reset

Type

int

Default value

0x0

SAU_REGION201.NSC

Set NSC for SAU region201 at reset

Type

bool

Default value

0x0

SAU_REGION202.BADDR

Base address of SAU region202 at reset

Type

int

Default value

0x0

SAU_REGION202.ENABLE

Enable SAU region202 at reset

Type

bool

Default value

0x0

SAU_REGION202.LADDR

Limit address of SAU region202 at reset

Type

int

Default value

0x0

SAU_REGION202.NSC

Set NSC for SAU region202 at reset

Type

bool

Default value

0x0

SAU_REGION203.BADDR

Base address of SAU region203 at reset

Type

int

Default value

0x0

SAU_REGION203.ENABLE

Enable SAU region203 at reset

Type

bool

Default value

0x0

SAU_REGION203.LADDR

Limit address of SAU region203 at reset

Type

int

Default value

0x0

SAU_REGION203.NSC

Set NSC for SAU region203 at reset

Type

bool

Default value

0x0

SAU_REGION204.BADDR

Base address of SAU region204 at reset

Type

int

Default value

0x0

SAU_REGION204.ENABLE

Enable SAU region204 at reset

Type

bool

Default value

0x0

SAU_REGION204.LADDR

Limit address of SAU region204 at reset

Type

int

Default value

0x0

SAU_REGION204.NSC

Set NSC for SAU region204 at reset

Type

bool

Default value

0x0

SAU_REGION205.BADDR

Base address of SAU region205 at reset

Type

int

Default value

0x0

SAU_REGION205.ENABLE

Enable SAU region205 at reset

Type

bool

Default value

0x0

SAU_REGION205.LADDR

Limit address of SAU region205 at reset

Type

int

Default value

0x0

SAU_REGION205.NSC

Set NSC for SAU region205 at reset

Type

bool

Default value

0x0

SAU_REGION206.BADDR

Base address of SAU region206 at reset

Type

int

Default value

0x0

SAU_REGION206.ENABLE

Enable SAU region206 at reset

Type

bool

Default value

0x0

SAU_REGION206.LADDR

Limit address of SAU region206 at reset

Type

int

Default value

0x0

SAU_REGION206.NSC

Set NSC for SAU region206 at reset

Type

bool

Default value

0x0

SAU_REGION207.BADDR

Base address of SAU region207 at reset

Type

int

Default value

0x0

SAU_REGION207.ENABLE

Enable SAU region207 at reset

Type

bool

Default value

0x0

SAU_REGION207.LADDR

Limit address of SAU region207 at reset

Type

int

Default value

0x0

SAU_REGION207.NSC

Set NSC for SAU region207 at reset

Type

bool

Default value

0x0

SAU_REGION208.BADDR

Base address of SAU region208 at reset

Type

int

Default value

0x0

SAU_REGION208.ENABLE

Enable SAU region208 at reset

Type

bool

Default value

0x0

SAU_REGION208.LADDR

Limit address of SAU region208 at reset

Type

int

Default value

0x0

SAU_REGION208.NSC

Set NSC for SAU region208 at reset

Type

bool

Default value

0x0

SAU_REGION209.BADDR

Base address of SAU region209 at reset

Type

int

Default value

0x0

SAU_REGION209.ENABLE

Enable SAU region209 at reset

Type

bool

Default value

0x0

SAU_REGION209.LADDR

Limit address of SAU region209 at reset

Type

int

Default value

0x0

SAU_REGION209.NSC

Set NSC for SAU region209 at reset

Type

bool

Default value

0x0

SAU_REGION21.BADDR

Base address of SAU region21 at reset

Type

int

Default value

0x0

SAU_REGION21.ENABLE

Enable SAU region21 at reset

Type

bool

Default value

0x0

SAU_REGION21.LADDR

Limit address of SAU region21 at reset

Type

int

Default value

0x0

SAU_REGION21.NSC

Set NSC for SAU region21 at reset

Type

bool

Default value

0x0

SAU_REGION210.BADDR

Base address of SAU region210 at reset

Type

int

Default value

0x0

SAU_REGION210.ENABLE

Enable SAU region210 at reset

Type

bool

Default value

0x0

SAU_REGION210.LADDR

Limit address of SAU region210 at reset

Type

int

Default value

0x0

SAU_REGION210.NSC

Set NSC for SAU region210 at reset

Type

bool

Default value

0x0

SAU_REGION211.BADDR

Base address of SAU region211 at reset

Type

int

Default value

0x0

SAU_REGION211.ENABLE

Enable SAU region211 at reset

Type

bool

Default value

0x0

SAU_REGION211.LADDR

Limit address of SAU region211 at reset

Type

int

Default value

0x0

SAU_REGION211.NSC

Set NSC for SAU region211 at reset

Type

bool

Default value

0x0

SAU_REGION212.BADDR

Base address of SAU region212 at reset

Type

int

Default value

0x0

SAU_REGION212.ENABLE

Enable SAU region212 at reset

Type

bool

Default value

0x0

SAU_REGION212.LADDR

Limit address of SAU region212 at reset

Type

int

Default value

0x0

SAU_REGION212.NSC

Set NSC for SAU region212 at reset

Type

bool

Default value

0x0

SAU_REGION213.BADDR

Base address of SAU region213 at reset

Type

int

Default value

0x0

SAU_REGION213.ENABLE

Enable SAU region213 at reset

Type

bool

Default value

0x0

SAU_REGION213.LADDR

Limit address of SAU region213 at reset

Type

int

Default value

0x0

SAU_REGION213.NSC

Set NSC for SAU region213 at reset

Type

bool

Default value

0x0

SAU_REGION214.BADDR

Base address of SAU region214 at reset

Type

int

Default value

0x0

SAU_REGION214.ENABLE

Enable SAU region214 at reset

Type

bool

Default value

0x0

SAU_REGION214.LADDR

Limit address of SAU region214 at reset

Type

int

Default value

0x0

SAU_REGION214.NSC

Set NSC for SAU region214 at reset

Type

bool

Default value

0x0

SAU_REGION215.BADDR

Base address of SAU region215 at reset

Type

int

Default value

0x0

SAU_REGION215.ENABLE

Enable SAU region215 at reset

Type

bool

Default value

0x0

SAU_REGION215.LADDR

Limit address of SAU region215 at reset

Type

int

Default value

0x0

SAU_REGION215.NSC

Set NSC for SAU region215 at reset

Type

bool

Default value

0x0

SAU_REGION216.BADDR

Base address of SAU region216 at reset

Type

int

Default value

0x0

SAU_REGION216.ENABLE

Enable SAU region216 at reset

Type

bool

Default value

0x0

SAU_REGION216.LADDR

Limit address of SAU region216 at reset

Type

int

Default value

0x0

SAU_REGION216.NSC

Set NSC for SAU region216 at reset

Type

bool

Default value

0x0

SAU_REGION217.BADDR

Base address of SAU region217 at reset

Type

int

Default value

0x0

SAU_REGION217.ENABLE

Enable SAU region217 at reset

Type

bool

Default value

0x0

SAU_REGION217.LADDR

Limit address of SAU region217 at reset

Type

int

Default value

0x0

SAU_REGION217.NSC

Set NSC for SAU region217 at reset

Type

bool

Default value

0x0

SAU_REGION218.BADDR

Base address of SAU region218 at reset

Type

int

Default value

0x0

SAU_REGION218.ENABLE

Enable SAU region218 at reset

Type

bool

Default value

0x0

SAU_REGION218.LADDR

Limit address of SAU region218 at reset

Type

int

Default value

0x0

SAU_REGION218.NSC

Set NSC for SAU region218 at reset

Type

bool

Default value

0x0

SAU_REGION219.BADDR

Base address of SAU region219 at reset

Type

int

Default value

0x0

SAU_REGION219.ENABLE

Enable SAU region219 at reset

Type

bool

Default value

0x0

SAU_REGION219.LADDR

Limit address of SAU region219 at reset

Type

int

Default value

0x0

SAU_REGION219.NSC

Set NSC for SAU region219 at reset

Type

bool

Default value

0x0

SAU_REGION22.BADDR

Base address of SAU region22 at reset

Type

int

Default value

0x0

SAU_REGION22.ENABLE

Enable SAU region22 at reset

Type

bool

Default value

0x0

SAU_REGION22.LADDR

Limit address of SAU region22 at reset

Type

int

Default value

0x0

SAU_REGION22.NSC

Set NSC for SAU region22 at reset

Type

bool

Default value

0x0

SAU_REGION220.BADDR

Base address of SAU region220 at reset

Type

int

Default value

0x0

SAU_REGION220.ENABLE

Enable SAU region220 at reset

Type

bool

Default value

0x0

SAU_REGION220.LADDR

Limit address of SAU region220 at reset

Type

int

Default value

0x0

SAU_REGION220.NSC

Set NSC for SAU region220 at reset

Type

bool

Default value

0x0

SAU_REGION221.BADDR

Base address of SAU region221 at reset

Type

int

Default value

0x0

SAU_REGION221.ENABLE

Enable SAU region221 at reset

Type

bool

Default value

0x0

SAU_REGION221.LADDR

Limit address of SAU region221 at reset

Type

int

Default value

0x0

SAU_REGION221.NSC

Set NSC for SAU region221 at reset

Type

bool

Default value

0x0

SAU_REGION222.BADDR

Base address of SAU region222 at reset

Type

int

Default value

0x0

SAU_REGION222.ENABLE

Enable SAU region222 at reset

Type

bool

Default value

0x0

SAU_REGION222.LADDR

Limit address of SAU region222 at reset

Type

int

Default value

0x0

SAU_REGION222.NSC

Set NSC for SAU region222 at reset

Type

bool

Default value

0x0

SAU_REGION223.BADDR

Base address of SAU region223 at reset

Type

int

Default value

0x0

SAU_REGION223.ENABLE

Enable SAU region223 at reset

Type

bool

Default value

0x0

SAU_REGION223.LADDR

Limit address of SAU region223 at reset

Type

int

Default value

0x0

SAU_REGION223.NSC

Set NSC for SAU region223 at reset

Type

bool

Default value

0x0

SAU_REGION224.BADDR

Base address of SAU region224 at reset

Type

int

Default value

0x0

SAU_REGION224.ENABLE

Enable SAU region224 at reset

Type

bool

Default value

0x0

SAU_REGION224.LADDR

Limit address of SAU region224 at reset

Type

int

Default value

0x0

SAU_REGION224.NSC

Set NSC for SAU region224 at reset

Type

bool

Default value

0x0

SAU_REGION225.BADDR

Base address of SAU region225 at reset

Type

int

Default value

0x0

SAU_REGION225.ENABLE

Enable SAU region225 at reset

Type

bool

Default value

0x0

SAU_REGION225.LADDR

Limit address of SAU region225 at reset

Type

int

Default value

0x0

SAU_REGION225.NSC

Set NSC for SAU region225 at reset

Type

bool

Default value

0x0

SAU_REGION226.BADDR

Base address of SAU region226 at reset

Type

int

Default value

0x0

SAU_REGION226.ENABLE

Enable SAU region226 at reset

Type

bool

Default value

0x0

SAU_REGION226.LADDR

Limit address of SAU region226 at reset

Type

int

Default value

0x0

SAU_REGION226.NSC

Set NSC for SAU region226 at reset

Type

bool

Default value

0x0

SAU_REGION227.BADDR

Base address of SAU region227 at reset

Type

int

Default value

0x0

SAU_REGION227.ENABLE

Enable SAU region227 at reset

Type

bool

Default value

0x0

SAU_REGION227.LADDR

Limit address of SAU region227 at reset

Type

int

Default value

0x0

SAU_REGION227.NSC

Set NSC for SAU region227 at reset

Type

bool

Default value

0x0

SAU_REGION228.BADDR

Base address of SAU region228 at reset

Type

int

Default value

0x0

SAU_REGION228.ENABLE

Enable SAU region228 at reset

Type

bool

Default value

0x0

SAU_REGION228.LADDR

Limit address of SAU region228 at reset

Type

int

Default value

0x0

SAU_REGION228.NSC

Set NSC for SAU region228 at reset

Type

bool

Default value

0x0

SAU_REGION229.BADDR

Base address of SAU region229 at reset

Type

int

Default value

0x0

SAU_REGION229.ENABLE

Enable SAU region229 at reset

Type

bool

Default value

0x0

SAU_REGION229.LADDR

Limit address of SAU region229 at reset

Type

int

Default value

0x0

SAU_REGION229.NSC

Set NSC for SAU region229 at reset

Type

bool

Default value

0x0

SAU_REGION23.BADDR

Base address of SAU region23 at reset

Type

int

Default value

0x0

SAU_REGION23.ENABLE

Enable SAU region23 at reset

Type

bool

Default value

0x0

SAU_REGION23.LADDR

Limit address of SAU region23 at reset

Type

int

Default value

0x0

SAU_REGION23.NSC

Set NSC for SAU region23 at reset

Type

bool

Default value

0x0

SAU_REGION230.BADDR

Base address of SAU region230 at reset

Type

int

Default value

0x0

SAU_REGION230.ENABLE

Enable SAU region230 at reset

Type

bool

Default value

0x0

SAU_REGION230.LADDR

Limit address of SAU region230 at reset

Type

int

Default value

0x0

SAU_REGION230.NSC

Set NSC for SAU region230 at reset

Type

bool

Default value

0x0

SAU_REGION231.BADDR

Base address of SAU region231 at reset

Type

int

Default value

0x0

SAU_REGION231.ENABLE

Enable SAU region231 at reset

Type

bool

Default value

0x0

SAU_REGION231.LADDR

Limit address of SAU region231 at reset

Type

int

Default value

0x0

SAU_REGION231.NSC

Set NSC for SAU region231 at reset

Type

bool

Default value

0x0

SAU_REGION232.BADDR

Base address of SAU region232 at reset

Type

int

Default value

0x0

SAU_REGION232.ENABLE

Enable SAU region232 at reset

Type

bool

Default value

0x0

SAU_REGION232.LADDR

Limit address of SAU region232 at reset

Type

int

Default value

0x0

SAU_REGION232.NSC

Set NSC for SAU region232 at reset

Type

bool

Default value

0x0

SAU_REGION233.BADDR

Base address of SAU region233 at reset

Type

int

Default value

0x0

SAU_REGION233.ENABLE

Enable SAU region233 at reset

Type

bool

Default value

0x0

SAU_REGION233.LADDR

Limit address of SAU region233 at reset

Type

int

Default value

0x0

SAU_REGION233.NSC

Set NSC for SAU region233 at reset

Type

bool

Default value

0x0

SAU_REGION234.BADDR

Base address of SAU region234 at reset

Type

int

Default value

0x0

SAU_REGION234.ENABLE

Enable SAU region234 at reset

Type

bool

Default value

0x0

SAU_REGION234.LADDR

Limit address of SAU region234 at reset

Type

int

Default value

0x0

SAU_REGION234.NSC

Set NSC for SAU region234 at reset

Type

bool

Default value

0x0

SAU_REGION235.BADDR

Base address of SAU region235 at reset

Type

int

Default value

0x0

SAU_REGION235.ENABLE

Enable SAU region235 at reset

Type

bool

Default value

0x0

SAU_REGION235.LADDR

Limit address of SAU region235 at reset

Type

int

Default value

0x0

SAU_REGION235.NSC

Set NSC for SAU region235 at reset

Type

bool

Default value

0x0

SAU_REGION236.BADDR

Base address of SAU region236 at reset

Type

int

Default value

0x0

SAU_REGION236.ENABLE

Enable SAU region236 at reset

Type

bool

Default value

0x0

SAU_REGION236.LADDR

Limit address of SAU region236 at reset

Type

int

Default value

0x0

SAU_REGION236.NSC

Set NSC for SAU region236 at reset

Type

bool

Default value

0x0

SAU_REGION237.BADDR

Base address of SAU region237 at reset

Type

int

Default value

0x0

SAU_REGION237.ENABLE

Enable SAU region237 at reset

Type

bool

Default value

0x0

SAU_REGION237.LADDR

Limit address of SAU region237 at reset

Type

int

Default value

0x0

SAU_REGION237.NSC

Set NSC for SAU region237 at reset

Type

bool

Default value

0x0

SAU_REGION238.BADDR

Base address of SAU region238 at reset

Type

int

Default value

0x0

SAU_REGION238.ENABLE

Enable SAU region238 at reset

Type

bool

Default value

0x0

SAU_REGION238.LADDR

Limit address of SAU region238 at reset

Type

int

Default value

0x0

SAU_REGION238.NSC

Set NSC for SAU region238 at reset

Type

bool

Default value

0x0

SAU_REGION239.BADDR

Base address of SAU region239 at reset

Type

int

Default value

0x0

SAU_REGION239.ENABLE

Enable SAU region239 at reset

Type

bool

Default value

0x0

SAU_REGION239.LADDR

Limit address of SAU region239 at reset

Type

int

Default value

0x0

SAU_REGION239.NSC

Set NSC for SAU region239 at reset

Type

bool

Default value

0x0

SAU_REGION24.BADDR

Base address of SAU region24 at reset

Type

int

Default value

0x0

SAU_REGION24.ENABLE

Enable SAU region24 at reset

Type

bool

Default value

0x0

SAU_REGION24.LADDR

Limit address of SAU region24 at reset

Type

int

Default value

0x0

SAU_REGION24.NSC

Set NSC for SAU region24 at reset

Type

bool

Default value

0x0

SAU_REGION240.BADDR

Base address of SAU region240 at reset

Type

int

Default value

0x0

SAU_REGION240.ENABLE

Enable SAU region240 at reset

Type

bool

Default value

0x0

SAU_REGION240.LADDR

Limit address of SAU region240 at reset

Type

int

Default value

0x0

SAU_REGION240.NSC

Set NSC for SAU region240 at reset

Type

bool

Default value

0x0

SAU_REGION241.BADDR

Base address of SAU region241 at reset

Type

int

Default value

0x0

SAU_REGION241.ENABLE

Enable SAU region241 at reset

Type

bool

Default value

0x0

SAU_REGION241.LADDR

Limit address of SAU region241 at reset

Type

int

Default value

0x0

SAU_REGION241.NSC

Set NSC for SAU region241 at reset

Type

bool

Default value

0x0

SAU_REGION242.BADDR

Base address of SAU region242 at reset

Type

int

Default value

0x0

SAU_REGION242.ENABLE

Enable SAU region242 at reset

Type

bool

Default value

0x0

SAU_REGION242.LADDR

Limit address of SAU region242 at reset

Type

int

Default value

0x0

SAU_REGION242.NSC

Set NSC for SAU region242 at reset

Type

bool

Default value

0x0

SAU_REGION243.BADDR

Base address of SAU region243 at reset

Type

int

Default value

0x0

SAU_REGION243.ENABLE

Enable SAU region243 at reset

Type

bool

Default value

0x0

SAU_REGION243.LADDR

Limit address of SAU region243 at reset

Type

int

Default value

0x0

SAU_REGION243.NSC

Set NSC for SAU region243 at reset

Type

bool

Default value

0x0

SAU_REGION244.BADDR

Base address of SAU region244 at reset

Type

int

Default value

0x0

SAU_REGION244.ENABLE

Enable SAU region244 at reset

Type

bool

Default value

0x0

SAU_REGION244.LADDR

Limit address of SAU region244 at reset

Type

int

Default value

0x0

SAU_REGION244.NSC

Set NSC for SAU region244 at reset

Type

bool

Default value

0x0

SAU_REGION245.BADDR

Base address of SAU region245 at reset

Type

int

Default value

0x0

SAU_REGION245.ENABLE

Enable SAU region245 at reset

Type

bool

Default value

0x0

SAU_REGION245.LADDR

Limit address of SAU region245 at reset

Type

int

Default value

0x0

SAU_REGION245.NSC

Set NSC for SAU region245 at reset

Type

bool

Default value

0x0

SAU_REGION246.BADDR

Base address of SAU region246 at reset

Type

int

Default value

0x0

SAU_REGION246.ENABLE

Enable SAU region246 at reset

Type

bool

Default value

0x0

SAU_REGION246.LADDR

Limit address of SAU region246 at reset

Type

int

Default value

0x0

SAU_REGION246.NSC

Set NSC for SAU region246 at reset

Type

bool

Default value

0x0

SAU_REGION247.BADDR

Base address of SAU region247 at reset

Type

int

Default value

0x0

SAU_REGION247.ENABLE

Enable SAU region247 at reset

Type

bool

Default value

0x0

SAU_REGION247.LADDR

Limit address of SAU region247 at reset

Type

int

Default value

0x0

SAU_REGION247.NSC

Set NSC for SAU region247 at reset

Type

bool

Default value

0x0

SAU_REGION248.BADDR

Base address of SAU region248 at reset

Type

int

Default value

0x0

SAU_REGION248.ENABLE

Enable SAU region248 at reset

Type

bool

Default value

0x0

SAU_REGION248.LADDR

Limit address of SAU region248 at reset

Type

int

Default value

0x0

SAU_REGION248.NSC

Set NSC for SAU region248 at reset

Type

bool

Default value

0x0

SAU_REGION249.BADDR

Base address of SAU region249 at reset

Type

int

Default value

0x0

SAU_REGION249.ENABLE

Enable SAU region249 at reset

Type

bool

Default value

0x0

SAU_REGION249.LADDR

Limit address of SAU region249 at reset

Type

int

Default value

0x0

SAU_REGION249.NSC

Set NSC for SAU region249 at reset

Type

bool

Default value

0x0

SAU_REGION25.BADDR

Base address of SAU region25 at reset

Type

int

Default value

0x0

SAU_REGION25.ENABLE

Enable SAU region25 at reset

Type

bool

Default value

0x0

SAU_REGION25.LADDR

Limit address of SAU region25 at reset

Type

int

Default value

0x0

SAU_REGION25.NSC

Set NSC for SAU region25 at reset

Type

bool

Default value

0x0

SAU_REGION250.BADDR

Base address of SAU region250 at reset

Type

int

Default value

0x0

SAU_REGION250.ENABLE

Enable SAU region250 at reset

Type

bool

Default value

0x0

SAU_REGION250.LADDR

Limit address of SAU region250 at reset

Type

int

Default value

0x0

SAU_REGION250.NSC

Set NSC for SAU region250 at reset

Type

bool

Default value

0x0

SAU_REGION251.BADDR

Base address of SAU region251 at reset

Type

int

Default value

0x0

SAU_REGION251.ENABLE

Enable SAU region251 at reset

Type

bool

Default value

0x0

SAU_REGION251.LADDR

Limit address of SAU region251 at reset

Type

int

Default value

0x0

SAU_REGION251.NSC

Set NSC for SAU region251 at reset

Type

bool

Default value

0x0

SAU_REGION252.BADDR

Base address of SAU region252 at reset

Type

int

Default value

0x0

SAU_REGION252.ENABLE

Enable SAU region252 at reset

Type

bool

Default value

0x0

SAU_REGION252.LADDR

Limit address of SAU region252 at reset

Type

int

Default value

0x0

SAU_REGION252.NSC

Set NSC for SAU region252 at reset

Type

bool

Default value

0x0

SAU_REGION253.BADDR

Base address of SAU region253 at reset

Type

int

Default value

0x0

SAU_REGION253.ENABLE

Enable SAU region253 at reset

Type

bool

Default value

0x0

SAU_REGION253.LADDR

Limit address of SAU region253 at reset

Type

int

Default value

0x0

SAU_REGION253.NSC

Set NSC for SAU region253 at reset

Type

bool

Default value

0x0

SAU_REGION254.BADDR

Base address of SAU region254 at reset

Type

int

Default value

0x0

SAU_REGION254.ENABLE

Enable SAU region254 at reset

Type

bool

Default value

0x0

SAU_REGION254.LADDR

Limit address of SAU region254 at reset

Type

int

Default value

0x0

SAU_REGION254.NSC

Set NSC for SAU region254 at reset

Type

bool

Default value

0x0

SAU_REGION255.BADDR

Base address of SAU region255 at reset

Type

int

Default value

0x0

SAU_REGION255.ENABLE

Enable SAU region255 at reset

Type

bool

Default value

0x0

SAU_REGION255.LADDR

Limit address of SAU region255 at reset

Type

int

Default value

0x0

SAU_REGION255.NSC

Set NSC for SAU region255 at reset

Type

bool

Default value

0x0

SAU_REGION26.BADDR

Base address of SAU region26 at reset

Type

int

Default value

0x0

SAU_REGION26.ENABLE

Enable SAU region26 at reset

Type

bool

Default value

0x0

SAU_REGION26.LADDR

Limit address of SAU region26 at reset

Type

int

Default value

0x0

SAU_REGION26.NSC

Set NSC for SAU region26 at reset

Type

bool

Default value

0x0

SAU_REGION27.BADDR

Base address of SAU region27 at reset

Type

int

Default value

0x0

SAU_REGION27.ENABLE

Enable SAU region27 at reset

Type

bool

Default value

0x0

SAU_REGION27.LADDR

Limit address of SAU region27 at reset

Type

int

Default value

0x0

SAU_REGION27.NSC

Set NSC for SAU region27 at reset

Type

bool

Default value

0x0

SAU_REGION28.BADDR

Base address of SAU region28 at reset

Type

int

Default value

0x0

SAU_REGION28.ENABLE

Enable SAU region28 at reset

Type

bool

Default value

0x0

SAU_REGION28.LADDR

Limit address of SAU region28 at reset

Type

int

Default value

0x0

SAU_REGION28.NSC

Set NSC for SAU region28 at reset

Type

bool

Default value

0x0

SAU_REGION29.BADDR

Base address of SAU region29 at reset

Type

int

Default value

0x0

SAU_REGION29.ENABLE

Enable SAU region29 at reset

Type

bool

Default value

0x0

SAU_REGION29.LADDR

Limit address of SAU region29 at reset

Type

int

Default value

0x0

SAU_REGION29.NSC

Set NSC for SAU region29 at reset

Type

bool

Default value

0x0

SAU_REGION3.BADDR

Base address of SAU region3 at reset

Type

int

Default value

0x0

SAU_REGION3.ENABLE

Enable SAU region3 at reset

Type

bool

Default value

0x0

SAU_REGION3.LADDR

Limit address of SAU region3 at reset

Type

int

Default value

0x0

SAU_REGION3.NSC

Set NSC for SAU region3 at reset

Type

bool

Default value

0x0

SAU_REGION30.BADDR

Base address of SAU region30 at reset

Type

int

Default value

0x0

SAU_REGION30.ENABLE

Enable SAU region30 at reset

Type

bool

Default value

0x0

SAU_REGION30.LADDR

Limit address of SAU region30 at reset

Type

int

Default value

0x0

SAU_REGION30.NSC

Set NSC for SAU region30 at reset

Type

bool

Default value

0x0

SAU_REGION31.BADDR

Base address of SAU region31 at reset

Type

int

Default value

0x0

SAU_REGION31.ENABLE

Enable SAU region31 at reset

Type

bool

Default value

0x0

SAU_REGION31.LADDR

Limit address of SAU region31 at reset

Type

int

Default value

0x0

SAU_REGION31.NSC

Set NSC for SAU region31 at reset

Type

bool

Default value

0x0

SAU_REGION32.BADDR

Base address of SAU region32 at reset

Type

int

Default value

0x0

SAU_REGION32.ENABLE

Enable SAU region32 at reset

Type

bool

Default value

0x0

SAU_REGION32.LADDR

Limit address of SAU region32 at reset

Type

int

Default value

0x0

SAU_REGION32.NSC

Set NSC for SAU region32 at reset

Type

bool

Default value

0x0

SAU_REGION33.BADDR

Base address of SAU region33 at reset

Type

int

Default value

0x0

SAU_REGION33.ENABLE

Enable SAU region33 at reset

Type

bool

Default value

0x0

SAU_REGION33.LADDR

Limit address of SAU region33 at reset

Type

int

Default value

0x0

SAU_REGION33.NSC

Set NSC for SAU region33 at reset

Type

bool

Default value

0x0

SAU_REGION34.BADDR

Base address of SAU region34 at reset

Type

int

Default value

0x0

SAU_REGION34.ENABLE

Enable SAU region34 at reset

Type

bool

Default value

0x0

SAU_REGION34.LADDR

Limit address of SAU region34 at reset

Type

int

Default value

0x0

SAU_REGION34.NSC

Set NSC for SAU region34 at reset

Type

bool

Default value

0x0

SAU_REGION35.BADDR

Base address of SAU region35 at reset

Type

int

Default value

0x0

SAU_REGION35.ENABLE

Enable SAU region35 at reset

Type

bool

Default value

0x0

SAU_REGION35.LADDR

Limit address of SAU region35 at reset

Type

int

Default value

0x0

SAU_REGION35.NSC

Set NSC for SAU region35 at reset

Type

bool

Default value

0x0

SAU_REGION36.BADDR

Base address of SAU region36 at reset

Type

int

Default value

0x0

SAU_REGION36.ENABLE

Enable SAU region36 at reset

Type

bool

Default value

0x0

SAU_REGION36.LADDR

Limit address of SAU region36 at reset

Type

int

Default value

0x0

SAU_REGION36.NSC

Set NSC for SAU region36 at reset

Type

bool

Default value

0x0

SAU_REGION37.BADDR

Base address of SAU region37 at reset

Type

int

Default value

0x0

SAU_REGION37.ENABLE

Enable SAU region37 at reset

Type

bool

Default value

0x0

SAU_REGION37.LADDR

Limit address of SAU region37 at reset

Type

int

Default value

0x0

SAU_REGION37.NSC

Set NSC for SAU region37 at reset

Type

bool

Default value

0x0

SAU_REGION38.BADDR

Base address of SAU region38 at reset

Type

int

Default value

0x0

SAU_REGION38.ENABLE

Enable SAU region38 at reset

Type

bool

Default value

0x0

SAU_REGION38.LADDR

Limit address of SAU region38 at reset

Type

int

Default value

0x0

SAU_REGION38.NSC

Set NSC for SAU region38 at reset

Type

bool

Default value

0x0

SAU_REGION39.BADDR

Base address of SAU region39 at reset

Type

int

Default value

0x0

SAU_REGION39.ENABLE

Enable SAU region39 at reset

Type

bool

Default value

0x0

SAU_REGION39.LADDR

Limit address of SAU region39 at reset

Type

int

Default value

0x0

SAU_REGION39.NSC

Set NSC for SAU region39 at reset

Type

bool

Default value

0x0

SAU_REGION4.BADDR

Base address of SAU region4 at reset

Type

int

Default value

0x0

SAU_REGION4.ENABLE

Enable SAU region4 at reset

Type

bool

Default value

0x0

SAU_REGION4.LADDR

Limit address of SAU region4 at reset

Type

int

Default value

0x0

SAU_REGION4.NSC

Set NSC for SAU region4 at reset

Type

bool

Default value

0x0

SAU_REGION40.BADDR

Base address of SAU region40 at reset

Type

int

Default value

0x0

SAU_REGION40.ENABLE

Enable SAU region40 at reset

Type

bool

Default value

0x0

SAU_REGION40.LADDR

Limit address of SAU region40 at reset

Type

int

Default value

0x0

SAU_REGION40.NSC

Set NSC for SAU region40 at reset

Type

bool

Default value

0x0

SAU_REGION41.BADDR

Base address of SAU region41 at reset

Type

int

Default value

0x0

SAU_REGION41.ENABLE

Enable SAU region41 at reset

Type

bool

Default value

0x0

SAU_REGION41.LADDR

Limit address of SAU region41 at reset

Type

int

Default value

0x0

SAU_REGION41.NSC

Set NSC for SAU region41 at reset

Type

bool

Default value

0x0

SAU_REGION42.BADDR

Base address of SAU region42 at reset

Type

int

Default value

0x0

SAU_REGION42.ENABLE

Enable SAU region42 at reset

Type

bool

Default value

0x0

SAU_REGION42.LADDR

Limit address of SAU region42 at reset

Type

int

Default value

0x0

SAU_REGION42.NSC

Set NSC for SAU region42 at reset

Type

bool

Default value

0x0

SAU_REGION43.BADDR

Base address of SAU region43 at reset

Type

int

Default value

0x0

SAU_REGION43.ENABLE

Enable SAU region43 at reset

Type

bool

Default value

0x0

SAU_REGION43.LADDR

Limit address of SAU region43 at reset

Type

int

Default value

0x0

SAU_REGION43.NSC

Set NSC for SAU region43 at reset

Type

bool

Default value

0x0

SAU_REGION44.BADDR

Base address of SAU region44 at reset

Type

int

Default value

0x0

SAU_REGION44.ENABLE

Enable SAU region44 at reset

Type

bool

Default value

0x0

SAU_REGION44.LADDR

Limit address of SAU region44 at reset

Type

int

Default value

0x0

SAU_REGION44.NSC

Set NSC for SAU region44 at reset

Type

bool

Default value

0x0

SAU_REGION45.BADDR

Base address of SAU region45 at reset

Type

int

Default value

0x0

SAU_REGION45.ENABLE

Enable SAU region45 at reset

Type

bool

Default value

0x0

SAU_REGION45.LADDR

Limit address of SAU region45 at reset

Type

int

Default value

0x0

SAU_REGION45.NSC

Set NSC for SAU region45 at reset

Type

bool

Default value

0x0

SAU_REGION46.BADDR

Base address of SAU region46 at reset

Type

int

Default value

0x0

SAU_REGION46.ENABLE

Enable SAU region46 at reset

Type

bool

Default value

0x0

SAU_REGION46.LADDR

Limit address of SAU region46 at reset

Type

int

Default value

0x0

SAU_REGION46.NSC

Set NSC for SAU region46 at reset

Type

bool

Default value

0x0

SAU_REGION47.BADDR

Base address of SAU region47 at reset

Type

int

Default value

0x0

SAU_REGION47.ENABLE

Enable SAU region47 at reset

Type

bool

Default value

0x0

SAU_REGION47.LADDR

Limit address of SAU region47 at reset

Type

int

Default value

0x0

SAU_REGION47.NSC

Set NSC for SAU region47 at reset

Type

bool

Default value

0x0

SAU_REGION48.BADDR

Base address of SAU region48 at reset

Type

int

Default value

0x0

SAU_REGION48.ENABLE

Enable SAU region48 at reset

Type

bool

Default value

0x0

SAU_REGION48.LADDR

Limit address of SAU region48 at reset

Type

int

Default value

0x0

SAU_REGION48.NSC

Set NSC for SAU region48 at reset

Type

bool

Default value

0x0

SAU_REGION49.BADDR

Base address of SAU region49 at reset

Type

int

Default value

0x0

SAU_REGION49.ENABLE

Enable SAU region49 at reset

Type

bool

Default value

0x0

SAU_REGION49.LADDR

Limit address of SAU region49 at reset

Type

int

Default value

0x0

SAU_REGION49.NSC

Set NSC for SAU region49 at reset

Type

bool

Default value

0x0

SAU_REGION5.BADDR

Base address of SAU region5 at reset

Type

int

Default value

0x0

SAU_REGION5.ENABLE

Enable SAU region5 at reset

Type

bool

Default value

0x0

SAU_REGION5.LADDR

Limit address of SAU region5 at reset

Type

int

Default value

0x0

SAU_REGION5.NSC

Set NSC for SAU region5 at reset

Type

bool

Default value

0x0

SAU_REGION50.BADDR

Base address of SAU region50 at reset

Type

int

Default value

0x0

SAU_REGION50.ENABLE

Enable SAU region50 at reset

Type

bool

Default value

0x0

SAU_REGION50.LADDR

Limit address of SAU region50 at reset

Type

int

Default value

0x0

SAU_REGION50.NSC

Set NSC for SAU region50 at reset

Type

bool

Default value

0x0

SAU_REGION51.BADDR

Base address of SAU region51 at reset

Type

int

Default value

0x0

SAU_REGION51.ENABLE

Enable SAU region51 at reset

Type

bool

Default value

0x0

SAU_REGION51.LADDR

Limit address of SAU region51 at reset

Type

int

Default value

0x0

SAU_REGION51.NSC

Set NSC for SAU region51 at reset

Type

bool

Default value

0x0

SAU_REGION52.BADDR

Base address of SAU region52 at reset

Type

int

Default value

0x0

SAU_REGION52.ENABLE

Enable SAU region52 at reset

Type

bool

Default value

0x0

SAU_REGION52.LADDR

Limit address of SAU region52 at reset

Type

int

Default value

0x0

SAU_REGION52.NSC

Set NSC for SAU region52 at reset

Type

bool

Default value

0x0

SAU_REGION53.BADDR

Base address of SAU region53 at reset

Type

int

Default value

0x0

SAU_REGION53.ENABLE

Enable SAU region53 at reset

Type

bool

Default value

0x0

SAU_REGION53.LADDR

Limit address of SAU region53 at reset

Type

int

Default value

0x0

SAU_REGION53.NSC

Set NSC for SAU region53 at reset

Type

bool

Default value

0x0

SAU_REGION54.BADDR

Base address of SAU region54 at reset

Type

int

Default value

0x0

SAU_REGION54.ENABLE

Enable SAU region54 at reset

Type

bool

Default value

0x0

SAU_REGION54.LADDR

Limit address of SAU region54 at reset

Type

int

Default value

0x0

SAU_REGION54.NSC

Set NSC for SAU region54 at reset

Type

bool

Default value

0x0

SAU_REGION55.BADDR

Base address of SAU region55 at reset

Type

int

Default value

0x0

SAU_REGION55.ENABLE

Enable SAU region55 at reset

Type

bool

Default value

0x0

SAU_REGION55.LADDR

Limit address of SAU region55 at reset

Type

int

Default value

0x0

SAU_REGION55.NSC

Set NSC for SAU region55 at reset

Type

bool

Default value

0x0

SAU_REGION56.BADDR

Base address of SAU region56 at reset

Type

int

Default value

0x0

SAU_REGION56.ENABLE

Enable SAU region56 at reset

Type

bool

Default value

0x0

SAU_REGION56.LADDR

Limit address of SAU region56 at reset

Type

int

Default value

0x0

SAU_REGION56.NSC

Set NSC for SAU region56 at reset

Type

bool

Default value

0x0

SAU_REGION57.BADDR

Base address of SAU region57 at reset

Type

int

Default value

0x0

SAU_REGION57.ENABLE

Enable SAU region57 at reset

Type

bool

Default value

0x0

SAU_REGION57.LADDR

Limit address of SAU region57 at reset

Type

int

Default value

0x0

SAU_REGION57.NSC

Set NSC for SAU region57 at reset

Type

bool

Default value

0x0

SAU_REGION58.BADDR

Base address of SAU region58 at reset

Type

int

Default value

0x0

SAU_REGION58.ENABLE

Enable SAU region58 at reset

Type

bool

Default value

0x0

SAU_REGION58.LADDR

Limit address of SAU region58 at reset

Type

int

Default value

0x0

SAU_REGION58.NSC

Set NSC for SAU region58 at reset

Type

bool

Default value

0x0

SAU_REGION59.BADDR

Base address of SAU region59 at reset

Type

int

Default value

0x0

SAU_REGION59.ENABLE

Enable SAU region59 at reset

Type

bool

Default value

0x0

SAU_REGION59.LADDR

Limit address of SAU region59 at reset

Type

int

Default value

0x0

SAU_REGION59.NSC

Set NSC for SAU region59 at reset

Type

bool

Default value

0x0

SAU_REGION6.BADDR

Base address of SAU region6 at reset

Type

int

Default value

0x0

SAU_REGION6.ENABLE

Enable SAU region6 at reset

Type

bool

Default value

0x0

SAU_REGION6.LADDR

Limit address of SAU region6 at reset

Type

int

Default value

0x0

SAU_REGION6.NSC

Set NSC for SAU region6 at reset

Type

bool

Default value

0x0

SAU_REGION60.BADDR

Base address of SAU region60 at reset

Type

int

Default value

0x0

SAU_REGION60.ENABLE

Enable SAU region60 at reset

Type

bool

Default value

0x0

SAU_REGION60.LADDR

Limit address of SAU region60 at reset

Type

int

Default value

0x0

SAU_REGION60.NSC

Set NSC for SAU region60 at reset

Type

bool

Default value

0x0

SAU_REGION61.BADDR

Base address of SAU region61 at reset

Type

int

Default value

0x0

SAU_REGION61.ENABLE

Enable SAU region61 at reset

Type

bool

Default value

0x0

SAU_REGION61.LADDR

Limit address of SAU region61 at reset

Type

int

Default value

0x0

SAU_REGION61.NSC

Set NSC for SAU region61 at reset

Type

bool

Default value

0x0

SAU_REGION62.BADDR

Base address of SAU region62 at reset

Type

int

Default value

0x0

SAU_REGION62.ENABLE

Enable SAU region62 at reset

Type

bool

Default value

0x0

SAU_REGION62.LADDR

Limit address of SAU region62 at reset

Type

int

Default value

0x0

SAU_REGION62.NSC

Set NSC for SAU region62 at reset

Type

bool

Default value

0x0

SAU_REGION63.BADDR

Base address of SAU region63 at reset

Type

int

Default value

0x0

SAU_REGION63.ENABLE

Enable SAU region63 at reset

Type

bool

Default value

0x0

SAU_REGION63.LADDR

Limit address of SAU region63 at reset

Type

int

Default value

0x0

SAU_REGION63.NSC

Set NSC for SAU region63 at reset

Type

bool

Default value

0x0

SAU_REGION64.BADDR

Base address of SAU region64 at reset

Type

int

Default value

0x0

SAU_REGION64.ENABLE

Enable SAU region64 at reset

Type

bool

Default value

0x0

SAU_REGION64.LADDR

Limit address of SAU region64 at reset

Type

int

Default value

0x0

SAU_REGION64.NSC

Set NSC for SAU region64 at reset

Type

bool

Default value

0x0

SAU_REGION65.BADDR

Base address of SAU region65 at reset

Type

int

Default value

0x0

SAU_REGION65.ENABLE

Enable SAU region65 at reset

Type

bool

Default value

0x0

SAU_REGION65.LADDR

Limit address of SAU region65 at reset

Type

int

Default value

0x0

SAU_REGION65.NSC

Set NSC for SAU region65 at reset

Type

bool

Default value

0x0

SAU_REGION66.BADDR

Base address of SAU region66 at reset

Type

int

Default value

0x0

SAU_REGION66.ENABLE

Enable SAU region66 at reset

Type

bool

Default value

0x0

SAU_REGION66.LADDR

Limit address of SAU region66 at reset

Type

int

Default value

0x0

SAU_REGION66.NSC

Set NSC for SAU region66 at reset

Type

bool

Default value

0x0

SAU_REGION67.BADDR

Base address of SAU region67 at reset

Type

int

Default value

0x0

SAU_REGION67.ENABLE

Enable SAU region67 at reset

Type

bool

Default value

0x0

SAU_REGION67.LADDR

Limit address of SAU region67 at reset

Type

int

Default value

0x0

SAU_REGION67.NSC

Set NSC for SAU region67 at reset

Type

bool

Default value

0x0

SAU_REGION68.BADDR

Base address of SAU region68 at reset

Type

int

Default value

0x0

SAU_REGION68.ENABLE

Enable SAU region68 at reset

Type

bool

Default value

0x0

SAU_REGION68.LADDR

Limit address of SAU region68 at reset

Type

int

Default value

0x0

SAU_REGION68.NSC

Set NSC for SAU region68 at reset

Type

bool

Default value

0x0

SAU_REGION69.BADDR

Base address of SAU region69 at reset

Type

int

Default value

0x0

SAU_REGION69.ENABLE

Enable SAU region69 at reset

Type

bool

Default value

0x0

SAU_REGION69.LADDR

Limit address of SAU region69 at reset

Type

int

Default value

0x0

SAU_REGION69.NSC

Set NSC for SAU region69 at reset

Type

bool

Default value

0x0

SAU_REGION7.BADDR

Base address of SAU region7 at reset

Type

int

Default value

0x0

SAU_REGION7.ENABLE

Enable SAU region7 at reset

Type

bool

Default value

0x0

SAU_REGION7.LADDR

Limit address of SAU region7 at reset

Type

int

Default value

0x0

SAU_REGION7.NSC

Set NSC for SAU region7 at reset

Type

bool

Default value

0x0

SAU_REGION70.BADDR

Base address of SAU region70 at reset

Type

int

Default value

0x0

SAU_REGION70.ENABLE

Enable SAU region70 at reset

Type

bool

Default value

0x0

SAU_REGION70.LADDR

Limit address of SAU region70 at reset

Type

int

Default value

0x0

SAU_REGION70.NSC

Set NSC for SAU region70 at reset

Type

bool

Default value

0x0

SAU_REGION71.BADDR

Base address of SAU region71 at reset

Type

int

Default value

0x0

SAU_REGION71.ENABLE

Enable SAU region71 at reset

Type

bool

Default value

0x0

SAU_REGION71.LADDR

Limit address of SAU region71 at reset

Type

int

Default value

0x0

SAU_REGION71.NSC

Set NSC for SAU region71 at reset

Type

bool

Default value

0x0

SAU_REGION72.BADDR

Base address of SAU region72 at reset

Type

int

Default value

0x0

SAU_REGION72.ENABLE

Enable SAU region72 at reset

Type

bool

Default value

0x0

SAU_REGION72.LADDR

Limit address of SAU region72 at reset

Type

int

Default value

0x0

SAU_REGION72.NSC

Set NSC for SAU region72 at reset

Type

bool

Default value

0x0

SAU_REGION73.BADDR

Base address of SAU region73 at reset

Type

int

Default value

0x0

SAU_REGION73.ENABLE

Enable SAU region73 at reset

Type

bool

Default value

0x0

SAU_REGION73.LADDR

Limit address of SAU region73 at reset

Type

int

Default value

0x0

SAU_REGION73.NSC

Set NSC for SAU region73 at reset

Type

bool

Default value

0x0

SAU_REGION74.BADDR

Base address of SAU region74 at reset

Type

int

Default value

0x0

SAU_REGION74.ENABLE

Enable SAU region74 at reset

Type

bool

Default value

0x0

SAU_REGION74.LADDR

Limit address of SAU region74 at reset

Type

int

Default value

0x0

SAU_REGION74.NSC

Set NSC for SAU region74 at reset

Type

bool

Default value

0x0

SAU_REGION75.BADDR

Base address of SAU region75 at reset

Type

int

Default value

0x0

SAU_REGION75.ENABLE

Enable SAU region75 at reset

Type

bool

Default value

0x0

SAU_REGION75.LADDR

Limit address of SAU region75 at reset

Type

int

Default value

0x0

SAU_REGION75.NSC

Set NSC for SAU region75 at reset

Type

bool

Default value

0x0

SAU_REGION76.BADDR

Base address of SAU region76 at reset

Type

int

Default value

0x0

SAU_REGION76.ENABLE

Enable SAU region76 at reset

Type

bool

Default value

0x0

SAU_REGION76.LADDR

Limit address of SAU region76 at reset

Type

int

Default value

0x0

SAU_REGION76.NSC

Set NSC for SAU region76 at reset

Type

bool

Default value

0x0

SAU_REGION77.BADDR

Base address of SAU region77 at reset

Type

int

Default value

0x0

SAU_REGION77.ENABLE

Enable SAU region77 at reset

Type

bool

Default value

0x0

SAU_REGION77.LADDR

Limit address of SAU region77 at reset

Type

int

Default value

0x0

SAU_REGION77.NSC

Set NSC for SAU region77 at reset

Type

bool

Default value

0x0

SAU_REGION78.BADDR

Base address of SAU region78 at reset

Type

int

Default value

0x0

SAU_REGION78.ENABLE

Enable SAU region78 at reset

Type

bool

Default value

0x0

SAU_REGION78.LADDR

Limit address of SAU region78 at reset

Type

int

Default value

0x0

SAU_REGION78.NSC

Set NSC for SAU region78 at reset

Type

bool

Default value

0x0

SAU_REGION79.BADDR

Base address of SAU region79 at reset

Type

int

Default value

0x0

SAU_REGION79.ENABLE

Enable SAU region79 at reset

Type

bool

Default value

0x0

SAU_REGION79.LADDR

Limit address of SAU region79 at reset

Type

int

Default value

0x0

SAU_REGION79.NSC

Set NSC for SAU region79 at reset

Type

bool

Default value

0x0

SAU_REGION8.BADDR

Base address of SAU region8 at reset

Type

int

Default value

0x0

SAU_REGION8.ENABLE

Enable SAU region8 at reset

Type

bool

Default value

0x0

SAU_REGION8.LADDR

Limit address of SAU region8 at reset

Type

int

Default value

0x0

SAU_REGION8.NSC

Set NSC for SAU region8 at reset

Type

bool

Default value

0x0

SAU_REGION80.BADDR

Base address of SAU region80 at reset

Type

int

Default value

0x0

SAU_REGION80.ENABLE

Enable SAU region80 at reset

Type

bool

Default value

0x0

SAU_REGION80.LADDR

Limit address of SAU region80 at reset

Type

int

Default value

0x0

SAU_REGION80.NSC

Set NSC for SAU region80 at reset

Type

bool

Default value

0x0

SAU_REGION81.BADDR

Base address of SAU region81 at reset

Type

int

Default value

0x0

SAU_REGION81.ENABLE

Enable SAU region81 at reset

Type

bool

Default value

0x0

SAU_REGION81.LADDR

Limit address of SAU region81 at reset

Type

int

Default value

0x0

SAU_REGION81.NSC

Set NSC for SAU region81 at reset

Type

bool

Default value

0x0

SAU_REGION82.BADDR

Base address of SAU region82 at reset

Type

int

Default value

0x0

SAU_REGION82.ENABLE

Enable SAU region82 at reset

Type

bool

Default value

0x0

SAU_REGION82.LADDR

Limit address of SAU region82 at reset

Type

int

Default value

0x0

SAU_REGION82.NSC

Set NSC for SAU region82 at reset

Type

bool

Default value

0x0

SAU_REGION83.BADDR

Base address of SAU region83 at reset

Type

int

Default value

0x0

SAU_REGION83.ENABLE

Enable SAU region83 at reset

Type

bool

Default value

0x0

SAU_REGION83.LADDR

Limit address of SAU region83 at reset

Type

int

Default value

0x0

SAU_REGION83.NSC

Set NSC for SAU region83 at reset

Type

bool

Default value

0x0

SAU_REGION84.BADDR

Base address of SAU region84 at reset

Type

int

Default value

0x0

SAU_REGION84.ENABLE

Enable SAU region84 at reset

Type

bool

Default value

0x0

SAU_REGION84.LADDR

Limit address of SAU region84 at reset

Type

int

Default value

0x0

SAU_REGION84.NSC

Set NSC for SAU region84 at reset

Type

bool

Default value

0x0

SAU_REGION85.BADDR

Base address of SAU region85 at reset

Type

int

Default value

0x0

SAU_REGION85.ENABLE

Enable SAU region85 at reset

Type

bool

Default value

0x0

SAU_REGION85.LADDR

Limit address of SAU region85 at reset

Type

int

Default value

0x0

SAU_REGION85.NSC

Set NSC for SAU region85 at reset

Type

bool

Default value

0x0

SAU_REGION86.BADDR

Base address of SAU region86 at reset

Type

int

Default value

0x0

SAU_REGION86.ENABLE

Enable SAU region86 at reset

Type

bool

Default value

0x0

SAU_REGION86.LADDR

Limit address of SAU region86 at reset

Type

int

Default value

0x0

SAU_REGION86.NSC

Set NSC for SAU region86 at reset

Type

bool

Default value

0x0

SAU_REGION87.BADDR

Base address of SAU region87 at reset

Type

int

Default value

0x0

SAU_REGION87.ENABLE

Enable SAU region87 at reset

Type

bool

Default value

0x0

SAU_REGION87.LADDR

Limit address of SAU region87 at reset

Type

int

Default value

0x0

SAU_REGION87.NSC

Set NSC for SAU region87 at reset

Type

bool

Default value

0x0

SAU_REGION88.BADDR

Base address of SAU region88 at reset

Type

int

Default value

0x0

SAU_REGION88.ENABLE

Enable SAU region88 at reset

Type

bool

Default value

0x0

SAU_REGION88.LADDR

Limit address of SAU region88 at reset

Type

int

Default value

0x0

SAU_REGION88.NSC

Set NSC for SAU region88 at reset

Type

bool

Default value

0x0

SAU_REGION89.BADDR

Base address of SAU region89 at reset

Type

int

Default value

0x0

SAU_REGION89.ENABLE

Enable SAU region89 at reset

Type

bool

Default value

0x0

SAU_REGION89.LADDR

Limit address of SAU region89 at reset

Type

int

Default value

0x0

SAU_REGION89.NSC

Set NSC for SAU region89 at reset

Type

bool

Default value

0x0

SAU_REGION9.BADDR

Base address of SAU region9 at reset

Type

int

Default value

0x0

SAU_REGION9.ENABLE

Enable SAU region9 at reset

Type

bool

Default value

0x0

SAU_REGION9.LADDR

Limit address of SAU region9 at reset

Type

int

Default value

0x0

SAU_REGION9.NSC

Set NSC for SAU region9 at reset

Type

bool

Default value

0x0

SAU_REGION90.BADDR

Base address of SAU region90 at reset

Type

int

Default value

0x0

SAU_REGION90.ENABLE

Enable SAU region90 at reset

Type

bool

Default value

0x0

SAU_REGION90.LADDR

Limit address of SAU region90 at reset

Type

int

Default value

0x0

SAU_REGION90.NSC

Set NSC for SAU region90 at reset

Type

bool

Default value

0x0

SAU_REGION91.BADDR

Base address of SAU region91 at reset

Type

int

Default value

0x0

SAU_REGION91.ENABLE

Enable SAU region91 at reset

Type

bool

Default value

0x0

SAU_REGION91.LADDR

Limit address of SAU region91 at reset

Type

int

Default value

0x0

SAU_REGION91.NSC

Set NSC for SAU region91 at reset

Type

bool

Default value

0x0

SAU_REGION92.BADDR

Base address of SAU region92 at reset

Type

int

Default value

0x0

SAU_REGION92.ENABLE

Enable SAU region92 at reset

Type

bool

Default value

0x0

SAU_REGION92.LADDR

Limit address of SAU region92 at reset

Type

int

Default value

0x0

SAU_REGION92.NSC

Set NSC for SAU region92 at reset

Type

bool

Default value

0x0

SAU_REGION93.BADDR

Base address of SAU region93 at reset

Type

int

Default value

0x0

SAU_REGION93.ENABLE

Enable SAU region93 at reset

Type

bool

Default value

0x0

SAU_REGION93.LADDR

Limit address of SAU region93 at reset

Type

int

Default value

0x0

SAU_REGION93.NSC

Set NSC for SAU region93 at reset

Type

bool

Default value

0x0

SAU_REGION94.BADDR

Base address of SAU region94 at reset

Type

int

Default value

0x0

SAU_REGION94.ENABLE

Enable SAU region94 at reset

Type

bool

Default value

0x0

SAU_REGION94.LADDR

Limit address of SAU region94 at reset

Type

int

Default value

0x0

SAU_REGION94.NSC

Set NSC for SAU region94 at reset

Type

bool

Default value

0x0

SAU_REGION95.BADDR

Base address of SAU region95 at reset

Type

int

Default value

0x0

SAU_REGION95.ENABLE

Enable SAU region95 at reset

Type

bool

Default value

0x0

SAU_REGION95.LADDR

Limit address of SAU region95 at reset

Type

int

Default value

0x0

SAU_REGION95.NSC

Set NSC for SAU region95 at reset

Type

bool

Default value

0x0

SAU_REGION96.BADDR

Base address of SAU region96 at reset

Type

int

Default value

0x0

SAU_REGION96.ENABLE

Enable SAU region96 at reset

Type

bool

Default value

0x0

SAU_REGION96.LADDR

Limit address of SAU region96 at reset

Type

int

Default value

0x0

SAU_REGION96.NSC

Set NSC for SAU region96 at reset

Type

bool

Default value

0x0

SAU_REGION97.BADDR

Base address of SAU region97 at reset

Type

int

Default value

0x0

SAU_REGION97.ENABLE

Enable SAU region97 at reset

Type

bool

Default value

0x0

SAU_REGION97.LADDR

Limit address of SAU region97 at reset

Type

int

Default value

0x0

SAU_REGION97.NSC

Set NSC for SAU region97 at reset

Type

bool

Default value

0x0

SAU_REGION98.BADDR

Base address of SAU region98 at reset

Type

int

Default value

0x0

SAU_REGION98.ENABLE

Enable SAU region98 at reset

Type

bool

Default value

0x0

SAU_REGION98.LADDR

Limit address of SAU region98 at reset

Type

int

Default value

0x0

SAU_REGION98.NSC

Set NSC for SAU region98 at reset

Type

bool

Default value

0x0

SAU_REGION99.BADDR

Base address of SAU region99 at reset

Type

int

Default value

0x0

SAU_REGION99.ENABLE

Enable SAU region99 at reset

Type

bool

Default value

0x0

SAU_REGION99.LADDR

Limit address of SAU region99 at reset

Type

int

Default value

0x0

SAU_REGION99.NSC

Set NSC for SAU region99 at reset

Type

bool

Default value

0x0

SAU_TYPE.SREGION

Number of SAU regions (0 => no SAU)

Type

int

Default value

0x10

SECEXT

Whether the ARMv8-M Security Extensions are included

Type

bool

Default value

0x1

SYST

Include SysTick timer functionality (0=Absent, 1=Secure only, 2=Secure and NS)

Type

int

Default value

0x2

SYST_CALIB_NS_reset

SYST_CALIB_NS reset value

Type

int

Default value

0x0

SYST_CALIB_reset

SYST_CALIB reset value

Type

int

Default value

0x0

VTOR_NS

NonSecure Vector Table Offset Register is writeable

Type

bool

Default value

0x1

VTOR_NS_MASK

Non-Secure VTOR write mask

Type

int

Default value

0xffffffff80

VTOR_S

Secure Vector Table Offset Register is writeable

Type

bool

Default value

0x1

VTOR_S_MASK

Secure VTOR write mask

Type

int

Default value

0xffffffff80

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

abort_unaligned_nonNormal

If true, UNPREDICTABLE accesses of device and strongly ordered memory abort; if false they are allowed.

Type

bool

Default value

0x1

aircr_iesb_is_writable

IS the AIRCR.IESB bit [5] writable?

Type

bool

Default value

0x1

aircr_iesb_reset

Set the AIRCR.IESB bit [5] after reset

Type

bool

Default value

0x0

allow_dap_writes_while_core_running

Debug writes are respected even while the core is running, i.e. the core does not have to be halted

Type

bool

Default value

0x1

allow_debug_monitor_with_in_flight_inst

Allow handling Debug Monitor exception with in-flight instructions

Type

bool

Default value

0x0

allow_stack_accesses_to_ppb_space

Allow stack accesses to PPB space

Type

bool

Default value

0x0

always_undefinstr_over_nocp

Only v8.0M. Always fault with UNDEFINSTR for undefined instructions that fall in CP space (don't check coprocessor status)

Type

bool

Default value

0x0

apply_prigroup_to_pending_tree

DEPRECATED: please use sep_sec_state_then_apply_prigroup_to_pending. Original description: Apply AIRCR.PRIGROUP to the pending and active trees (instead of just the active tree) when selecting the highest priority pending exception

Type

bool

Default value

0x0

baseline

Use the baseline profile (if false, use mainline)

Type

bool

Default value

0x1

bp_on_2nd_halfword

Respect DWT/BPU breakpoint-hit on 2nd halfword of 32-bit instruction

Type

bool

Default value

0x1

callee_register_push_low_to_high

If true, push callee registers in order from R4 to R11. If false, push R11 to R4

Type

bool

Default value

0x1

cde_fp_check_on_unsupported

Run FP checks on both supported and unsupported CDE instructions

Type

bool

Default value

0x0

cde_impl_name

Name of the CDE implementation for this core (implementation contributed by MTI plugin).

Type

string

Default value**clear_non_secure_EXC_RETURN.ES_on_tailchain**

Clear EXC_RETURN.ES in LR value on entry to a tail-chained exception when returning from Non-secure state

Type

bool

Default value

0x1

condition_flags_reset

Reset Value of condition flags in APSR

Type

int

Default value

0x0

cpi_div

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_can_access_debug_regs

The DWT, BPU, ROM table, DCB, and the SHCSR and DFSR registers access from the processor

Type

bool

Default value

0x1

dbg_coproc_load_store_enable

Enable LDCX and STCX instructions

Type

int

Default value

0x0

dcache-invalidate-ns-cleans-s

Whether V8M DCI* in non-secure should clean-and-invalidate secure cache contents.

Type

bool

Default value

0x0

dcache-size

L1 D-cache size in bytes

Type

int

Default value

0x8000

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

delay_faultmask_update

Delay FAULTMASK update to context sync

Type

bool

Default value

0x0

delay_sysreg_update

Delay some system register updates (e.g. SHCSR) to context sync

Type

bool

Default value

0x0

do_exclusive_monitor_check_first

In exclusive stores, check local exclusive monitor before detecting other memory aborts

Type

bool

Default value

0x0

drop_mem_fault

Whether to drop mem_fault in favour of subsequently generated NOCP/secure fault in PushStack

Type

bool

Default value

0x0

dtcm_enable

Enable DTCM at reset

Type

bool

Default value

0x0

dtcm_size

DTCM size in KB

Type

int

Default value

0x100

dwt_unaligned_word_access_as_half_word

DWT Treat unaligned word access as half word or bytes.

Type

bool

Default value

0x1

enable_helium_extension

Enable Helium extension

Type

bool

Default value

0x0

exercise_strex_fail

Reject a pseudo-random majority of exclusive store instructions

Type

bool

Default value

0x0

has_ahbp

Are Vendor-Sys accesses sent to a separate bus (AHBP on CM7).

Type

bool

Default value

0x1

has_arm_v8-1m

Enable v8.1M architecture version and features

Type

bool

Default value

0x0

has_cde

Enables Custom Datapath Extensions

Type

bool

Default value

0x0

has_core_dside_bus_gasket

STL gasket enabled

Type

bool

Default value

0x0

has_lob_cache

Support for LOB cache (only if support for LO instructions is enabled as well).

Type

bool

Default value

0x1

has_m55_tcmcr

If true, enables the CortexM55 TCM Control Registers (ITCMCR and DTCMCR), If false, CortexM55 TCM Control Registers are disabled

Type

bool

Default value

0x0

has_pmu

Availability of optional PMU.

Type

bool

Default value

0x0

has_separate_etm_reset

If true, signal 'etmreset' resets the core, else the core power-on-reset does

Type

bool

Default value

0x0

has_unprivileged_debug

Unprivileged Debug Extension supported for Mainline Extension

Type

bool

Default value

0x1

has_writebuffer

Implement write accesses buffering before L1 cache. May affect ext_abort behaviour.

Type

bool

Default value

0x0

icache-size

L1 I-cache size in bytes

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

ignore_RNR_top_nibble

If set, only the bottom four bits of MPU_RNR.REGION are used

Type

bool

Default value

0x0

ignore_demcr_sdme_for_nonhalting_bkpt

Ignore the SDME bit of the DEMCR register when escalating a Debug Monitor exception to a HardFault.

Type

bool

Default value

0x0

ignore_out_of_range_RNR_write

If an MPU_RNR.REGION write is out of range, ignore it ; if false, MPU_RNR values wrap

Type

bool

Default value

0x0

ignore_unpred_SBZSBO

Use smaller decoder does not UNDEF some unpredictable SBZ/SBO fields.

Type

bool

Default value

0x0

ignore_unpred_ZeroRegistersInList

VLDM,VSTM,STM,LDM with no registers NOP instead of UNDEF.

Type

bool

Default value

0x0

itcm_enable

Enable ITCM at reset

Type

bool

Default value

0x0

itcm_size

ITCM size in KB

Type

int

Default value

0x100

late_arrival

Enable late arrival support

Type

bool

Default value

0x1

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

mve_has_atomic_ticks

Enable atomic ticks behaviour for vector instructions flagged as such (e.g. VLDR)

Type

bool

Default value

0x0

num_pmu_counters

Number of available PMU counters.

Type

int

Default value

0x1f

number_of_itm_stimulus_ports

The number of ITM stimulus ports

Type

int

Default value

0x20

pend_overriden_exception_on_stack_push

Mark any overridden exceptions on stack push as pending (instead of dropping them)

Type

bool

Default value

0x0

preserve_unknown_caller_save_regs_at_S_to_S

preserve unknown caller registers when they become UNKNOWN at secure to secure

Type

bool

Default value

0x1

ras_ERRFR0

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR1

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR10

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR11

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR12

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR13

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR14

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR15

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR16

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR17

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR18

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR19

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR2

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR20

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR21

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR22

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR23

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR24

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR25

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR26

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR27

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR28

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR29

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR3

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR30

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR31

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR32

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR33

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR34

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR35

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR36

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR37

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR38

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR39

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR4

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR40

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR41

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR42

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR43

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR44

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR45

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR46

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR47

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR48

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR49

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR5

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR50

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR51

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR52

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR53

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR54

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR55

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR6

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR7

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR8

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_ERRFR9

A JSON object or array of objects for each field of ERRFR. Records not described default to RAZ e.g. '{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'.

Type

string

Default value

'{"ED":0x2, "UI":0x2, "FI":0x2, "UE":0x2, "CFI":0x2, "CEC":0x2, "RP":0x1, "DUI":0x0, "CEO":0x1, "CI":0x2}'

ras_cei_pin

RAS: Critical error interrupt pin.

Type

int

Default value

0x2

ras_cei_support

RAS: Whether Critical Error Interrupt is supported

Type

bool

Default value

0x1

ras_eri_pin

RAS: Error recovery interrupt pin.

Type

int

Default value

0x1

ras_eri_support

RAS: Whether Error Recovery Interrupt is supported

Type

bool

Default value

0x1

ras_error_record

56 bit value that specifies which nodes out of 0-55 are implemented (ERRDEVID is derived from this parameter)

Type

int

Default value

0xffffffffffffffff

ras_fhi_pin

RAS: Fault handling interrupt pin.

Type

int

Default value

0x0

ras_fhi_support

RAS: Whether Fault Handling Interrupt is supported

Type

bool

Default value

0x1

rd_ns_bus_err_behave

External read aborts in nonsecure domain 0:ignored, 1:precise, 2:imprecise, 3=imprecise except SO.

Type

int

Default value

0x1

rd_s_bus_err_behave

External read aborts in secure domain 0:ignored, 1:precise, 2:imprecise, 3=imprecise except SO.

Type

int

Default value

0x1

register_reset_data

Data used to fill register bits when they become UNKNOWN at reset.

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

sep_sec_state_then_apply_prigroup_to_pending

Use separate comparison trees for Secure and Non-Secure pending exceptions and apply AIRCR.PRIGROUP to the output of each before they are compared to determine the overall highest priority

Type

bool

Default value

0x0

sequential_security_transitions

Allow transition of security state in sequential instruction fetches that cross from non-secure to secure memory with SG instruction 0: never, 1: always, 2: 32-bit instrs, 3: ISB.

Type

int

Default value

0x1

share_fault_address_reg

If true, Fault Address Register is shared

Type

bool

Default value

0x0

stack_limit_check

Implementation defined stack limit checks for instructions. Bit 0: Load-exclusive, Bit 1: Load-acquire, Bit 2: VLDM. Any instruction that can't be configured does stack limit check by default.

Type

int

Default value

0x7

stack_limit_check_optimization

Stack limit check optimization (0: limit check done for each word on the stack, 1: limit check done only on stack pointer)

Type

bool

Default value

0x1

stacking_writes_are_precise

Faults on stack writes are precise

Type

bool

Default value

0x1

supports_unprivileged

Enable support for Unprivileged/Privileged Extension

Type

bool

Default value

0x1

tail_chain

Enable tail-chaining optimisation

Type

bool

Default value

0x1

trace_style

MVE instruction trace style: Add 16 for [**-] beat trace. Add 32 for tracing IMPLIED LOB instructions. Add 64 to change opcode of implied BF to 0xBF00

Type

int

Default value

0x2

unknown_regs_at_exception_value

Data used to fill registers when they become UNKNOWN at exception and exception-return.

Type

int

Default value

0x0

unpred_WriteBackandBaseInList_stores_old_base_value

allow STM with write back to base register in register list

Type

bool

Default value

0x0

unpred_mon_step_write

Behavior on unpredictable updates to MON_STEP bit of DEMCR. 0: ignore write, 1: set one, 2: set zero

Type

int

Default value

0x0

unpred_msr_psr_with_one_mask_and_nodsp_is_nop

If true, MSR to *PSR with a one mask and no DSP does nothing.

Type

bool

Default value

0x1

unpred_msr_psr_with_zero_mask_is_nop

If true, MSR to *PSR with a zero mask does nothing.

Type

bool

Default value

0x0

unstack_R_regs_before_fp_cp_check

In exception return unstack normal register before checking fp coprocessor is enable to unstack FP register

Type

bool

Default value

0x0

vector_fetch_as_wpt_event

Watchpoint on exception vector fetch

Type

bool

Default value

0x0

vector_fetch_busfault_sets_HFSR_FORCED

Only v8.0M. Set HFSR.FORCED when a vector table read generates a HardFault

Type

bool

Default value

0x0

vector_fetch_on_ide

Perform vector fetch on I-side

Type

bool

Default value

0x1

vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

wr_ns_bus_err_behave

External write aborts in nonsecure domain 0:ignored, 1:precise, 2:imprecise, 3=imprecise except SO.

Type

int

Default value

0x3

wr_s_bus_err_behave

External write aborts in secure domain 0:ignored, 1:precise, 2:imprecise, 3=imprecise except SO.

Type

int

Default value

0x3

write_unknown_regs_at_exception

Do we write registers when they become UNKNOWN at exception or exception-return.

Type

bool

Default value

0x0

3.5.4 ARMCortexA5CT

ARMCortexA5CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-186: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA5CT

This model has the following Iris instances:

Table 3-187: ARMCortexA5CT Iris instances

InstanceName	ComponentName
ARMCortexA5CT	Cluster_ARM_Cortex-A5UP
ARMCortexA5CT.acp_mapper	PVBusMapper
ARMCortexA5CT.cpu0	ARM_Cortex-A5UP
ARMCortexA5CT.cpu0.UTLB	TLB
ARMCortexA5CT.cpu0.l1dcache	PVCache
ARMCortexA5CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA5CT.cpu0.l1icache	PVCache
ARMCortexA5CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA5CT.cpu0.utlb	TlbCadi
ARMCortexA5CT.ext_bus	PVBusLogger
ARMCortexA5CT.ext_bus.mapper	PVBusMapper
ARMCortexA5CT.l1_incoherent_interconnect	PVCache
ARMCortexA5CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave

InstanceName	ComponentName
ARMCortexA5CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexA5CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-188: ARMCortexA5CT MTI instances

InstanceName	ComponentName
ARMCortexA5CT.acp_mapper	PVBusMapper
ARMCortexA5CT.cpu0	ARM_Cortex-A5
ARMCortexA5CT.cpu0.UTLB	TLB
ARMCortexA5CT.cpu0.l1dcache	PVCache
ARMCortexA5CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA5CT.cpu0.l1icache	PVCache
ARMCortexA5CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA5CT.ext_bus	PVBusLogger
ARMCortexA5CT.ext_bus.mapper	PVBusMapper
ARMCortexA5CT.l1_incoherent_interconnect	PVCache
ARMCortexA5CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexA5CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexA5CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA5CT contains the following CADI targets:

- ARM_Cortex-A5UP
- Cluster_ARM_Cortex-A5UP
- PVCache
- TlbCadi

About ARMCortexA5CT

- The `ase-present` and `vfp-present` parameters configure the synthesis options:
 - vfp present and ase present**
Neon™ and VFPv3-D32 are supported.
 - vfp present and ase not present**
VFPv3-D16 is supported.
 - vfp not present and ase present**
Illegal. Forces `vfp-present` to true so model has Neon and VFPv3-D32 support.
 - vfp not present and ase not present**
Model has neither Neon nor VFPv3-D32 support.
- Specify `false` for the `device-accurate-tlb` parameter to enable modeling a different number of TLBs if this improves simulation performance. In this case, the simulation is architecturally-accurate, but not device-accurate. Architectural accuracy is almost always sufficient. Specify `true` if device accuracy is required.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- The RR bit in the SCTLR is ignored.
- The Power Control Register in the system control coprocessor is implemented but writing to it does not change the behavior of the model.
- This component implements L1 cache as architecturally defined, but does not implement L2 cache. If you require an L2 cache you can add a PL310 Level 2 Cache Controller component.

Ports for ARMCortexA5CT

Table 3-189: Ports

Name	Protocol	Type	Description
<code>cfgend[1]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgnmfi[1]</code>	Signal	Slave	This signal disables FIQ mask in CPSR.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>clusterid</code>	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
<code>cp15sdisable[1]</code>	Signal	Slave	This signal disables write access to some system control processor registers.
<code>event</code>	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
<code>fiq[1]</code>	Signal	Slave	This signal drives the CPU's fast-interrupt handling.

Name	Protocol	Type	Description
irq[1]	Signal	Slave	This signal drives the CPU's interrupt handling.
pmuirq[1]	Signal	Master	Interrupt signal from performance monitoring unit.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
reset[1]	Signal	Slave	Raising this signal will put the core into reset mode.
standbywfe[1]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[1]	Signal	Master	This signal indicates if a core is in WFI state.
teinit[1]	Signal	Slave	This signal provides default exception handling state.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vinithi[1]	Signal	Slave	This signal controls of the location of the exception vectors at reset.

Parameters for ARMCortexA5CT

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Initialize to BE8 endianness

Type

bool

Default value

0x0

cpuX.CFGNMFI

Enable nonmaskable FIQ interrupts on startup

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.POWERCTLI

Default power control state for processor

Type

int

Default value

0x0

cpuX.TEINIT

T32 exception enable. The default has exceptions including reset handled in A32 state

Type

bool

Default value

0x0

cpuX.VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpuX.ase-present

Set whether model has NEON support

Type

bool

Default value

0x1

cpuX.dcache-size

Set D-cache size in bytes

Type

int

Default value

0x8000

cpuX.icache-size

Set I-cache size in bytes

Type

int

Default value

0x8000

cpuX.min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

cpuX.semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

cpuX.semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

cpuX.vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_access_latency

D-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

D-Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

device-accurate-tlb

Specify whether all TLBs are modeled

Type

bool

Default value

0x0

icache-hit_latency

I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_access_latency

I-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

3.5.5 ARMCortexA5MPx1CT

ARMCortexA5MPx1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-190: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA5MPx1CT

This model has the following Iris instances:

Table 3-191: ARMCortexA5MPx1CT Iris instances

InstanceName	ComponentName
ARMCortexA5MPx1CT	Cluster_ARM_Cortex-A5MP
ARMCortexA5MPx1CT.acp_mapper	PVBusMapper
ARMCortexA5MPx1CT.cpu0	ARM_Cortex-A5MP
ARMCortexA5MPx1CT.cpu0.UTLB	TLB
ARMCortexA5MPx1CT.cpu0.l1dcache	PVCache
ARMCortexA5MPx1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA5MPx1CT.cpu0.l1icache	PVCache
ARMCortexA5MPx1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA5MPx1CT.cpu0.utlb	TlbCadi
ARMCortexA5MPx1CT.ext_bus	PVBusLogger
ARMCortexA5MPx1CT.ext_bus.mapper	PVBusMapper
ARMCortexA5MPx1CT.internal_shareability_remapper	PVBusMapper
ARMCortexA5MPx1CT.l1_incoherent_interconnect	PVCache
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave

InstanceName	ComponentName
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexA5MPx1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-192: ARMCortexA5MPx1CT MTI instances

InstanceName	ComponentName
ARMCortexA5MPx1CT.acp_mapper	PVBusMapper
ARMCortexA5MPx1CT.cpu0	ARM_Cortex-A5MP
ARMCortexA5MPx1CT.cpu0.UTLB	TLB
ARMCortexA5MPx1CT.cpu0.l1dcache	PVCache
ARMCortexA5MPx1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA5MPx1CT.cpu0.l1icache	PVCache
ARMCortexA5MPx1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA5MPx1CT.ext_bus	PVBusLogger
ARMCortexA5MPx1CT.ext_bus.mapper	PVBusMapper
ARMCortexA5MPx1CT.internal_shareability_remapper	PVBusMapper
ARMCortexA5MPx1CT.l1_incoherent_interconnect	PVCache
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave

InstanceName	ComponentName
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexA5MPx1CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexA5MPx1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA5MPx1CT contains the following CADI targets:

- ARM_Cortex-A5MP
- Cluster_ARM_Cortex-A5MP
- PVCache
- TlbCadi

About ARMCortexA5MPx1CT

- The following components also exist:
 - ARMCortexA5MPx2CT.
 - ARMCortexA5MPx4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

- The `ase-present` and `vfp-present` parameters configure the synthesis options:

vfp present and ase present

Neon™ and VFPv3-D32 are supported.

vfp present and ase not present

VFPv3-D16 is supported.

vfp not present and ase present

Illegal. Forces `vfp-present` to true so model has Neon and VFPv3-D32 support.

vfp not present and ase not present

Model has neither Neon nor VFPv3-D32 support.

- Specify `false` for the `device-accurate-tlb` parameter to enable modeling a different number of TLBs if this improves simulation performance. In this case, the simulation is architecturally-accurate, but not device-accurate. Architectural accuracy is almost always sufficient. Specify `true` if device accuracy is required.
- If you are using the `ARMCortexA5MPxnCT` component in a VE platform model, the `PERIPHBASE` parameter is set automatically to `0x2C000000` and is not visible in the parameter list.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- There is a single memory port combining instruction, data, DMA and peripheral access.
- The GIC does not respect the CFGSDISABLE signal. This leads to some registers being accessible when they must not be.
- The SCU enable bit is ignored. The SCU is always enabled.
- The SCU ignores the invalidate all register.
- The Broadcast TLB or cache operations in this model do not cause other cores in the cluster that are asleep because of *Wait For Interrupt* (WFI) to wake up.
- The RR bit in the SCTLR is ignored.
- The Power Control Register in the system control coprocessor is implemented but writing to it does not change the behavior of the model.
- When modeling the SCU, coherency operations are represented by a memory write followed by a read to refill from memory, rather than using cache-to-cache transfers.
- This component implements L1 cache as architecturally defined, but does not implement L2 cache. If you require an L2 cache you can add a PL310 Level 2 Cache Controller component.

Ports for ARMCortexA5MPx1CT

Table 3-193: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
cfgend[1]	Signal	Slave	This signal is for EE bit initialisation.
cfgnmfi[1]	Signal	Slave	This signal disables FIQ mask in CPSR.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
cp15sdisable[1]	Signal	Slave	This signal disables write access to some system control processor registers.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
filteren	Signal	Slave	This signal enables filtering of address ranges between master bus ports.
filterend	Value	Slave	This port sets end of region mapped to pvbus_m1.
filterstart	Value	Slave	This port sets start of region mapped to pvbus_m1.
fiq[1]	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
fiqout[1]	Signal	Master	This signal exports internal VGIC FIQ signal to the CPU.
ints[224]	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
irq[1]	Signal	Slave	This signal drives the CPU's interrupt handling.
irqout[1]	Signal	Master	This signal exports internal VGIC IRQ signal to the CPU.
periphbase	Value	Slave	This port sets the base address of private peripheral region.

Name	Protocol	Type	Description
periphclk_in	ClockSignal	Slave	The timer and the watchdog take need a clk that is scaled down at least by factor of two.
periphreset	Signal	Slave	This signal resets timer and interrupt controller.
pmuirq[1]	Signal	Master	Interrupt signal from performance monitoring unit.
pvbus_m0	PVBus	Master	AXI master 0 bus master channel.
pvbus_m1	PVBus	Master	AXI master 1 bus master channel.
pwrctl[1]	Value	Slave	This port sets reset value for scu CPU status register.
pwrctl0[1]	Value	Master	This port sends scu CPU status register bits.
reset[1]	Signal	Slave	Raising this signal will put the core into reset mode.
scureset	Signal	Slave	This signal resets SCU.
smpnamp[1]	Signal	Master	This signals AMP or SMP mode for each Cortex-A5 processor.
standbywfe[1]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[1]	Signal	Master	This signal indicates if a core is in WFI state.
teinit[1]	Signal	Slave	This signal provides default exception handling state.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vinithi[1]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
wdreset[1]	Signal	Slave	This signal resets individual watchdog.
wdresetreq[1]	Signal	Master	This signal resets rest of the CA5MP system.

Parameters for ARM Cortex A5MPx1CT

CFGSDISABLE

Disable some accesses to GIC registers

Type

bool

Default value

0x0

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

FILTEREN

Enable filtering of accesses through pvbus_m0

Type

bool

Default value

0x0

FILTEREND

End of region filtered to pvbus_m0

Type

int

Default value

0x0

FILTERSTART

Base of region filtered to pvbus_m0

Type

int

Default value

0x0

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Initialize to BE8 endianness

Type

bool

Default value

0x0

cpuX.CFGNMFI

Enable nonmaskable FIQ interrupts on startup

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.POWERCTLI

Default power control state for processor

Type

int

Default value

0x0

cpuX.SMPnAMP

Set whether the processor is part of a coherent domain

Type

bool

Default value

0x0

cpuX.TEINIT

T32 exception enable. The default has exceptions including reset handled in A32 state

Type

bool

Default value

0x0

cpuX.VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpuX.ase-present

Set whether model has NEON support

Type

bool

Default value

0x1

cpuX.dcache-size

Set D-cache size in bytes

Type

int

Default value

0x8000

cpuX.icache-size

Set I-cache size in bytes

Type

int

Default value

0x8000

cpuX.min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

cpuX.semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

cpuX.semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

cpuX.vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_access_latency

D-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

D-Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

device-accurate-tlb

Specify whether all TLBs are modeled

Type

bool

Default value

0x0

dic-spi_count

Number of shared peripheral interrupts implemented

Type

int

Default value

0x40

icache-hit_latency

I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_access_latency

I-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

3.5.6 ARMCortexA7x1CT

ARMCortexA7x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-194: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA7x1CT

This model has the following Iris instances:

Table 3-195: ARMCortexA7x1CT Iris instances

InstanceName	ComponentName
ARMCortexA7x1CT	Cluster_ARM_Cortex-A7
ARMCortexA7x1CT.Cortex-A7_GIC	GICv2
ARMCortexA7x1CT.acp_mapper	PVBusMapper
ARMCortexA7x1CT.cpu0	ARM_Cortex-A7
ARMCortexA7x1CT.cpu0.DTLB	TLB
ARMCortexA7x1CT.cpu0.ITLB	TLB
ARMCortexA7x1CT.cpu0.dtlb	TlbCadi
ARMCortexA7x1CT.cpu0.itlb	TlbCadi
ARMCortexA7x1CT.cpu0.l1dcache	PVCache
ARMCortexA7x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA7x1CT.cpu0.l1icache	PVCache
ARMCortexA7x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA7x1CT.ext_bus	PVBusLogger
ARMCortexA7x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA7x1CT.l2_cache	PVCache
ARMCortexA7x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[7]	PVBusSlave

InstanceName	ComponentName
ARMCortexA7x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA7x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-196: ARMCortexA7x1CT MTI instances

InstanceName	ComponentName
ARMCortexA7x1CT.Cortex-A7_GIC	GICv2
ARMCortexA7x1CT.acp_mapper	PVBusMapper
ARMCortexA7x1CT.cpu0	ARM_Cortex-A7
ARMCortexA7x1CT.cpu0.DTLB	TLB
ARMCortexA7x1CT.cpu0.ITLB	TLB
ARMCortexA7x1CT.cpu0.l1dcache	PVCache
ARMCortexA7x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA7x1CT.cpu0.l1icache	PVCache
ARMCortexA7x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA7x1CT.ext_bus	PVBusLogger
ARMCortexA7x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA7x1CT.l2_cache	PVCache
ARMCortexA7x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA7x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA7x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA7x1CT contains the following CADI targets:

- ARM_Cortex-A7
- Cluster_ARM_Cortex-A7
- PVMCache
- TlbCadi

About ARMCortexA7x1CT

- The following components also exist:
 - ARMCortexA7x2CT.
 - ARMCortexA7x3CT.
 - ARMCortexA7x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

- The `ase-present` and `vfp-present` parameters configure the synthesis options:

vfp present and ase present

Neon™ and VFPv4-D32 are supported.

vfp present and ase not present

VFPv4-D16 is supported.

vfp not present and ase present

Illegal. Forces `vfp-present` to true so model has Neon and VFPv4-D32 support.

vfp not present and ase not present

Model has neither Neon nor VFPv4-D32 support.

- If you are using the ARMCortexA7x_nCT component in a VE platform model, the `PERIPHBASE` parameter is set automatically to `0x2C000000` and is not visible in the parameter list.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.

ACE limitation

AXI Coherency Extensions (ACE) are extensions to AXI4 that support system-level cache coherency between multiple clusters. The ACE cache models in the Arm® Cortex®-A15 and Cortex-A7, and the ACE support in the CCI-400 have the limitation that they only process one transaction at a time. Normally, the simulation processes each transaction to completion before allowing any master to generate another transaction. However, in the following situation the simulation might fail. If a SystemC bus slave calls `wait()` while it is processing a transaction, this call might allow another master to issue another transaction that passes through the CCI-400 or the Cortex-A15 or Cortex-A7 caches. This situation could happen if a SystemC bus master running in another thread is connected to one of the ACE-lite ports on the CCI-400.

Differences between the CT model and RTL implementations

This model has the following differences from the corresponding revision of the RTL implementation:

- The GIC does not respect the `CFGSDISABLE` signal. This leads to some registers wrongly being accessible.

- The Broadcast Translation Lookaside Buffer (TLB) or cache operations in this model do not cause other cores in the cluster that are asleep because of Wait For Interrupt (WFI) to wake up.
- The model ignores the RR bit in the SCTLR.
- The model implements the Power Control Register in the system control coprocessor but writing to it does not change the behavior of the model.
- The model does not implement ETM registers.
- The model does not support the Cortex®-A7 mechanism to read the internal memory that the Cache and TLB structures use through the implementation defined region of the system coprocessor interface.
- The model does not upgrade DCIMVAC operations to DCCIMVAC.

Ports for ARMCortexA7x1CT

Table 3-197: Ports

Name	Protocol	Type	Description
axierrirq	Signal	Master	Imprecise aborts from the L2 are signaled by pulsing this pin, typically they are connect to an interrupt controller.
broadcastcachemaint	Signal	Slave	Enable broadcasting of cache maintenance operations to downstream caches.
broadcastinner	Signal	Slave	Enable broadcasting of Inner Shareable transactions.
broadcastouter	Signal	Slave	Enable broadcasting of Outer Shareable transactions.
cfgend[4]	Signal	Slave	This signal controls the SCTLR.EE bit.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clusterid	Value	Slave	This port sets the value in the CLUSTERID field (bits[11:8]) of the MPIDR.
CNTHPIRQ[4]	Signal	Master	Outputs of the generic timers.
CNTPNSIRQ[4]	Signal	Master	Outputs of the generic timers.
CNTPSIRQ[4]	Signal	Master	Outputs of the generic timers.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Outputs of the generic timers.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some secure system control processor registers.
cpuporeset[4]	Signal	Slave	Signal initializes all processor logic including NEON, VFP, Debug, PTM, breakpoint and watchpoint.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
fiqout[4]	Signal	Master	This signal exports internal VGIC FIQ signal to the CPU..
irq[4]	Signal	Slave	This signal drives the CPU's interrupt handling.
irqout[4]	Signal	Master	This signal exports internal VGIC IRQ signal to the CPU.
irqs[480]	Signal	Slave	These signals drive the CPU's interrupt controller interrupt lines.
l2reset	Signal	Slave	This signal resets the shared L2 memory system, interrupt controller and timer logic.
periphbase	Value_64	Slave	This port sets the base address of the private peripheral region.

Name	Protocol	Type	Description
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Signal initializes the shared Debug APB, CTI and CTM logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state.
standbywfil2	Signal	Master	Indicate that all the individual processors and the L2 memory system are in a WFI state.
teinit[4]	Signal	Slave	This signal enables Thumb exceptions (controls the SCTLR.TE bit).
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vfiq[4]	Signal	Slave	Virtual FIQ inputs. Note that the fiq pins are wired directly to the core if there is no internal VGIC. If there is an internal VGIC then these are ignored.
vinithi[4]	Signal	Slave	This signal controls the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtual IRQ inputs. Note that the irq pins are wired directly to the core if there is no internal VGIC. If there is an internal VGIC then these are ignored.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).

Parameters for ARM Cortex A7x1CT

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CFGSDISABLE

Disable some accesses to GIC registers

Type

bool

Default value

0x0

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Initialize to BE8 endianness

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.DBGROMADDR

This value is used to initialize the CP15 DBGDRAR register. Bits[39:12] of this register specify the ROM table physical address

Type

int

Default value

0x12000003

cpuX.DBGROMADDRV

If true this sets bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x1

cpuX.DBGSELFADDR

This value is used to initialize the CP15 DBGDSAR register. Bits[39:17] of this register specify the ROM table physical address

Type

int

Default value

0x10003

cpuX.DBGSELFADDRV

If true this sets bits[1:0] of the CP15 DBGDSAR to indicate that the address is valid

Type

bool

Default value

0x1

cpuX.TEINIT

T32 exception enable. The default has exceptions including reset handled in A32 state

Type

bool

Default value

0x0

cpuX.VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpuX.ase-present

Set whether CT model has been built with NEON support

Type

bool

Default value

0x1

cpuX.l1_dcache-size

Size of L1 D-cache

Type

int

Default value

0x8000

cpuX.l1_ichache-size

Size of L1 I-cache

Type

int

Default value

0x8000

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

cpuX.semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value

cpuX.semihosting-enable

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

cpuX.semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

cpuX.vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether CT model has been built with VFP support

Type

bool

Default value

0x1

dic-spi_count

Number of shared peripheral interrupts implemented

Type

int

Default value

0x40

disable_periph_decoder

Disable memory mapped access to gic system registers

Type

bool

Default value

0x0

internal_vgic

Configures whether the model of the processor contains a VGIC

Type

bool

Default value

0x1

l1_dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. `l1_dcache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l1_dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l1_dcache-state_modelled

Set whether L1 D-cache has stateful implementation

Type

bool

Default value

0x0

l1_dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `l1_dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l1_dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. `l1_dcache-write_access_latency` must be set to 0 for per-byte latencies to be applied. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l1_icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. l1_icache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l1_ichache-state_modelled

Set whether L1 I-cache has stateful implementation

Type

bool

Default value

0x0

l2_cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `l2_cache-read_latency` is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2_cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-size

Set L2 cache size in bytes

Type

int

Default value

0x80000

l2_cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-state_modelled

Set whether L2 cache has stateful implementation

Type

bool

Default value

0x0

l2_cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2_cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2_cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

3.5.7 ARMCortexA8CT

ARMCortexA8CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-198: IP revisions support

Revision	Quality level
r2p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA8CT

This model has the following Iris instances:

Table 3-199: ARMCortexA8CT Iris instances

InstanceName	ComponentName
ARMCortexA8CT	ARM_Cortex-A8
ARMCortexA8CT.DTLB	TLB

InstanceName	ComponentName
ARMCortexA8CT.ITLB	TLB
ARMCortexA8CT.acp_mapper	PVBusMapper
ARMCortexA8CT.cpu0.dtlb	TlbCadi
ARMCortexA8CT.cpu0.itlb	TlbCadi
ARMCortexA8CT.cpu0.l1dcache	PVCache
ARMCortexA8CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA8CT.cpu0.l1licache	PVCache
ARMCortexA8CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA8CT.ext_bus	PVBusLogger
ARMCortexA8CT.ext_bus.mapper	PVBusMapper
ARMCortexA8CT.l2_cache	PVCache
ARMCortexA8CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA8CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-200: ARMCortexA8CT MTI instances

InstanceName	ComponentName
ARMCortexA8CT	ARM_Cortex-A8
ARMCortexA8CT.DTLB	TLB
ARMCortexA8CT.ITLB	TLB
ARMCortexA8CT.acp_mapper	PVBusMapper
ARMCortexA8CT.cpu0.l1dcache	PVCache
ARMCortexA8CT.cpu0.l1dcache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA8CT.cpu0.l1icache	PVCache
ARMCortexA8CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA8CT.ext_bus	PVBusLogger
ARMCortexA8CT.ext_bus.mapper	PVBusMapper
ARMCortexA8CT.l2_cache	PVCache
ARMCortexA8CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA8CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA8CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA8CT contains the following CADI targets:

- ARM_Cortex-A8
- PVCache
- TlbCadi

About ARMCortexA8CT

- Specify `false` for the `device-accurate-tlb` parameter to enable modeling a different number of TLBs if this improves simulation performance. In this case, the simulation is architecturally-accurate, but not device-accurate. Architectural accuracy is almost always sufficient. Specify `true` if device accuracy is required.
- If the `l1_dcache-state_modelled` parameter is `true`, then `l2_cache-state_modelled` must also be `true`.
- When you use the `semihosting-cmd_line` parameter to set the command line that is available to semihosting SVC calls, the value of `argv[0]` is set to the first command-line argument, not to the name of an image.

- This component provides the registers that the Technical Reference Manual (TRM) specifies, except for the coprocessor 14 registers, the integration and test registers, and the PLE model, which is register-based and has no implemented behavior.

These TLB registers do not have working implementations:

- D-TLB ATTR read/write.
- D-TLB CAM read/write.
- D-TLB PA read/write.
- Normal memory remap register.
- Primary memory remap register.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- There is a single memory port combining instruction, data, DMA and peripheral access.
- The L2 cache write allocate policy is not configurable. It defaults to write-allocate. Writes to the configuration register succeed but are ignored, meaning that data can be unexpectedly stored in the L2 cache.
- Unaligned accesses with the MMU disabled on the processor do not cause data aborts.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.
- ECC and parity schemes are hardware-specific so are not supported.

Ports for ARMCortexA8CT

Table 3-201: Ports

Name	Protocol	Type	Description
<code>cfgend0</code>	Signal	Slave	Configure BE8 mode after a reset.
<code>cfgnmfi</code>	Signal	Slave	Configure FIQs as non-maskable after a reset.
<code>cfgte</code>	Signal	Slave	Configure exceptions to be taken in thumb mode after a reset.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>dmaexterrirq</code>	Signal	Master	L1 PLE error interrupt.
<code>dmairq</code>	Signal	Master	Interrupt signal from L1 PLE.
<code>dmairq</code>	Signal	Master	Secure interrupt signal from L1 PLE.
<code>fiq</code>	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
<code>irq</code>	Signal	Slave	This signal drives the CPU's interrupt handling.
<code>pmuirq</code>	Signal	Master	Interrupt signal from performance monitoring unit.
<code>pvbus_m</code>	PVBus	Master	The core will generate bus requests on this port.
<code>reset</code>	Signal	Slave	Raising this signal will put the core into reset mode.
<code>ticks</code>	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.
<code>vinithi</code>	Signal	Slave	Configure high vectors after a reset.

Parameters for ARMCortexA8CT

CFGEND0

Initialize to BE8 endianness

Type

bool

Default value

0x0

CFGNMFI

Enable nonmaskable FIQ interrupts on startup

Type

bool

Default value

0x0

CFGTE

Initialize to take exceptions in T32 state. Model starts in T32 state

Type

bool

Default value

0x0

CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

device-accurate-tlb

Specify whether all TLBs are modeled

Type

bool

Default value

0x0

implements_vfp

Set whether the model has been built with VFP and NEON support

Type

bool

Default value

0x1

l1_dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. l1_dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-size

Set L1 D-cache size in bytes

Type

int

Default value

0x8000

l1_dcache-state_modelled

Include Level 1 data cache state model

Type

bool

Default value

0x0

l1_dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. l1_dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l1_icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `l1_icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l1_icache-size

Set L1 I-cache size in bytes

Type

int

Default value

0x8000

l1_icache-state_modelled

Include Level 1 instruction cache state model

Type

bool

Default value

0x0

l2_cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2_cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2_cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-size

Set L2 cache size in bytes

Type

int

Default value

0x40000

l2_cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-state_modelled

Include unified Level 2 cache state model

Type

bool

Default value

0x0

l2_cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2_cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2_cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

semihosting-stack_base
Virtual address of base of descending stack

Type
int

Default value
0x10000000

semihosting-stack_limit
Virtual address of stack limit

Type
int

Default value
0xf000000

siliconID
Value as read by the system coprocessor siliconID register

Type
int

Default value
0x41000000

vfp-enable_at_reset
Enable coprocessor access and VFP at reset

Type
bool

Default value
0x0

3.5.8 ARMCortexA9MPx1CT

ARMCortexA9MPx1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-202: IP revisions support

Revision	Quality level
r3p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA9MPx1CT

This model has the following Iris instances:

Table 3-203: ARMCortexA9MPx1CT Iris instances

InstanceName	ComponentName
ARMCortexA9MPx1CT	Cluster_ARM_Cortex-A9MP
ARMCortexA9MPx1CT.acp_mapper	PVBusMapper
ARMCortexA9MPx1CT.cpu0	ARM_Cortex-A9MP
ARMCortexA9MPx1CT.cpu0.UTLB	TLB
ARMCortexA9MPx1CT.cpu0.l1dcache	PVCache
ARMCortexA9MPx1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA9MPx1CT.cpu0.l1icache	PVCache
ARMCortexA9MPx1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA9MPx1CT.cpu0.utlb	TlbCadi
ARMCortexA9MPx1CT.ext_bus	PVBusLogger
ARMCortexA9MPx1CT.ext_bus.mapper	PVBusMapper
ARMCortexA9MPx1CT.internal_shareability_remapper	PVBusMapper
ARMCortexA9MPx1CT.l1_incoherent_interconnect	PVCache
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexA9MPx1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-204: ARMCortexA9MPx1CT MTI instances

InstanceName	ComponentName
ARMCortexA9MPx1CT.acp_mapper	PVBusMapper
ARMCortexA9MPx1CT.cpu0	ARM_Cortex-A9MP

InstanceName	ComponentName
ARMCortexA9MPx1CT.cpu0.UTLB	TLB
ARMCortexA9MPx1CT.cpu0.l1dcache	PVCache
ARMCortexA9MPx1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA9MPx1CT.cpu0.l1icache	PVCache
ARMCortexA9MPx1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA9MPx1CT.ext_bus	PVBusLogger
ARMCortexA9MPx1CT.ext_bus.mapper	PVBusMapper
ARMCortexA9MPx1CT.internal_shareability_remapper	PVBusMapper
ARMCortexA9MPx1CT.l1_incoherent_interconnect	PVCache
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexA9MPx1CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexA9MPx1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA9MPx1CT contains the following CADI targets:

- ARM_Cortex-A9MP
- Cluster_ARM_Cortex-A9MP
- PVCache
- TlbCadi

About ARMCortexA9MPx1CT

- The following components also exist:
 - ARMCortexA9MPx2CT.
 - ARMCortexA9MPx4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

- The `ase-present` and `vfp-present` parameters configure the synthesis options:

vfp present and ase present

Neon™ and VFPv3-D32 are supported.

vfp present and ase not present

VFPv3-D16 is supported.

vfp not present and ase present

Illegal. Forces `vfp-present` to true so model has Neon and VFPv3-D32 support.

vfp not present and ase not present

Model has neither Neon nor VFPv3-D32 support.

- If you are using the `ARMCortexA9MPxnCT` component in a VE platform model, the `PERIPHBASE` parameter is set automatically to `0x2C000000` and is not visible in the parameter list.
- When you use the `semihosting-cmd_line` parameter to set the command line that is available to semihosting SVC calls, the value of `argv[0]` is set to the first command-line argument, not to the name of an image.
- Specify `false` for the `device-accurate-tlb` parameter to enable modeling a different number of TLBs if this improves simulation performance. In this case, the simulation is architecturally-accurate, but not device-accurate. Architectural accuracy is almost always sufficient. Specify `true` if device accuracy is required.
- This component implements L1 cache as architecturally defined, but does not implement L2 cache. If you require L2 cache, you can add a PL310 Level 2 Cache Controller component.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- The GIC does not respect the `CFGSDISABLE` signal. This leads to some registers being accessible when they must not be.
- The SCU enable bit is ignored. The SCU is always enabled.
- The SCU ignores the invalidate all register.
- The Broadcast TLB or cache operations in this model do not cause other cores in the cluster that are asleep because of WFI to wake up.
- The RR bit in the SCTL_R is ignored.
- The Power Control Register in the system control coprocessor is implemented but writing to it does not change the behavior of the model.
- The model cannot be configured with a 128-entry TLB.
- When modeling the SCU, coherency operations are represented by a memory write followed by a read to refill from memory, rather than using cache-to-cache transfers.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.
- The following TLB registers do not have working implementations:

- Main TLB Attr.
- Main TLB PA.
- Main TLB VA.
- Normal memory remap register.
- Primary memory remap register.
- Read Main TLB Entry.
- Write Main TLB Entry.

In addition, the simulation does not distinguish peripheral accesses from data accesses, so it ignores configuration of the peripheral port memory remap register.

- Parity error support is hardware-specific so is not modeled.

Ports for ARMCortexA9MPx1CT

Table 3-205: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
cfgend[4]	Signal	Slave	This signal is for EE bit initialisation.
cfgnmfi[4]	Signal	Slave	This signal disables FIQ mask in CPSR.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
filteren	Signal	Slave	This signal enables filtering of address ranges between master bus ports.
filterend	Value	Slave	This port sets end of region mapped to pvbus_m1.
filterstart	Value	Slave	This port sets start of region mapped to pvbus_m1.
fiq[4]	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
fiqout[4]	Signal	Master	This signal exports internal VGIC FIQ signal to the CPU..
ints[224]	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
irq[4]	Signal	Slave	This signal drives the CPU's interrupt handling.
irqout[4]	Signal	Master	This signal exports internal VGIC IRQ signal to the CPU.
periphbase	Value	Slave	This port sets the base address of private peripheral region.
periphclk_in	ClockSignal	Slave	The timer and the watchdog take need a clk that is scaled down at least by factor of two.
periphreset	Signal	Slave	This signal resets timer and interrupt controller.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
pvbus_m0	PVBus	Master	AXI master 0 bus master channel.
pvbus_m1	PVBus	Master	AXI master 1 bus master channel.
pwrcctl[4]	Value	Slave	This port sets reset value for scu CPU status register.

Name	Protocol	Type	Description
pwrtclo[4]	Value	Master	This port sends scu CPU status register bits.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
scureset	Signal	Slave	This signal resets SCU.
smpnamp[4]	Signal	Master	This signals AMP or SMP mode for each Cortex-A9 processor.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state.
teinit[4]	Signal	Slave	This signal provides default exception handling state.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
wdreset[4]	Signal	Slave	This signal resets individual watchdog.
wdresetreq[4]	Signal	Master	This signal resets rest of the CA9MP system.

Parameters for ARMCortexA9MPx1CT

CFGSDISABLE

Disable some accesses to GIC registers

Type

bool

Default value

0x0

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

FILTEREN

Enable filtering of accesses through pvbus_m0

Type

bool

Default value

0x0

FILTEREND

End of region filtered to pvbus_m0

Type

int

Default value

0x0

FILTERSTART

Base of region filtered to pvbus_m0

Type

int

Default value

0x0

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Initialize to BE8 endianness

Type

bool

Default value

0x0

cpuX.CFGNMFI

Enable nonmaskable FIQ interrupts on startup

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.POWERCTLI

Default power control state for processor

Type

int

Default value

0x0

cpuX.SMPnAMP

Set whether the processor is part of a coherent domain

Type

bool

Default value

0x0

cpuX.TEINIT

T32 exception enable. The default has exceptions including reset handled in A32 state

Type

bool

Default value

0x0

cpuX.VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpuX.ase-present

Set whether model has NEON support

Type

bool

Default value

0x1

cpuX.dcache-size

Set D-cache size in bytes

Type

int

Default value

0x8000

cpuX.icache-size

Set I-cache size in bytes

Type

int

Default value

0x8000

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

cpuX.semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

cpuX.semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

cpuX.vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

D-cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

D-cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

D-cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_access_latency

D-cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

D-cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

D-cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

D-cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

device-accurate-tlb

Specify whether all TLBs are modeled

Type

bool

Default value

0x0

dic-spi_count

Number of shared peripheral interrupts implemented

Type

int

Default value

0x40

icache-hit_latency

I-cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

I-cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

I-cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_access_latency

I-cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

I-cache timing annotation latency for read accesses given in ticks per byte accessed.icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

3.5.9 ARMCortexA9UPCT

ARMCortexA9UPCT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-206: IP revisions support

Revision	Quality level
r3p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARM Cortex-A9UPCT

This model has the following Iris instances:

Table 3-207: ARM Cortex-A9UPCT Iris instances

InstanceName	ComponentName
ARM Cortex-A9UPCT	Cluster_ARM_Cortex-A9UP
ARM Cortex-A9UPCT.acp_mapper	PVBusMapper
ARM Cortex-A9UPCT.cpu0	ARM_Cortex-A9UP
ARM Cortex-A9UPCT.cpu0.UTLB	TLB
ARM Cortex-A9UPCT.cpu0.l1dcache	PVCache
ARM Cortex-A9UPCT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM Cortex-A9UPCT.cpu0.l1icache	PVCache
ARM Cortex-A9UPCT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM Cortex-A9UPCT.cpu0.utlb	TlbCadi
ARM Cortex-A9UPCT.ext_bus	PVBusLogger
ARM Cortex-A9UPCT.ext_bus.mapper	PVBusMapper
ARM Cortex-A9UPCT.l1_incoherent_interconnect	PVCache
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARM Cortex-A9UPCT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARM Cortex-A9UPCT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-208: ARMCortexA9UPCT MTI instances

InstanceName	ComponentName
ARMCortexA9UPCT.acp_mapper	PVBusMapper
ARMCortexA9UPCT.cpu0	ARM_Cortex-A9UP
ARMCortexA9UPCT.cpu0.UTLB	TLB
ARMCortexA9UPCT.cpu0.l1dcache	PVCache
ARMCortexA9UPCT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA9UPCT.cpu0.l1icache	PVCache
ARMCortexA9UPCT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA9UPCT.ext_bus	PVBusLogger
ARMCortexA9UPCT.ext_bus.mapper	PVBusMapper
ARMCortexA9UPCT.l1_incoherent_interconnect	PVCache
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexA9UPCT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexA9UPCT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA9UPCT contains the following CADI targets:

- ARM_Cortex-A9UP
- Cluster_ARM_Cortex-A9UP
- PVCache
- TlbCadi

About ARMCortexA9UPCT

- The `ase-present` and `vfp-present` parameters configure the synthesis options:

vfp present and ase present

Neon™ and VFPv3-D32 are supported.

vfp present and ase not present

VFPv3-D16 is supported.

vfp not present and ase present

Illegal. Forces `vfp-present` to `true` so model has Neon and VFPv3-D32 support.

vfp not present and ase not present

Model has neither Neon nor VFPv3-D32 support.

- Specify `false` for the `device-accurate-tlb` parameter to enable modeling a different number of TLBs if this improves simulation performance. In this case, the simulation is architecturally-accurate, but not device-accurate. Architectural accuracy is almost always sufficient. Specify `true` if device accuracy is required.
- When you use the `semihosting-cmd_line` parameter to set the command line that is available to semihosting SVC calls, the value of `argv[0]` is set to the first command-line argument, not to the name of an image.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- The RR bit in the SCTLR is ignored.
- The Power Control Register in the system control coprocessor is implemented but writing to it does not change the behavior of the model.
- The model cannot be configured with a 128-entry TLB.
- This component implements L1 cache as architecturally defined, but does not implement L2 cache. If you require L2 cache you can add a PL310 Level 2 Cache Controller component.
- Parity error support is hardware-specific so is not modeled.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.

These TLB registers do not have working implementations:

- Normal memory remap register.
- Primary memory remap register.
- Read Main TLB Entry.
- Write Main TLB Entry.
- Main TLB VA.
- Main TLB PA.
- Main TLB Attr.

In addition, the simulation does not distinguish peripheral accesses from data accesses, so it ignores configuration of the peripheral port memory remap register.

Ports for ARMCortexA9UPCT

Table 3-209: Ports

Name	Protocol	Type	Description
cfgend[1]	Signal	Slave	This signal is for EE bit initialisation.
cfgnmfi[1]	Signal	Slave	This signal disables FIQ mask in CPSR.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
cp15sdisable[1]	Signal	Slave	This signal disables write access to some system control processor registers.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[1]	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
irq[1]	Signal	Slave	This signal drives the CPU's interrupt handling.
pmuirq[1]	Signal	Master	Interrupt signal from performance monitoring unit.
pvbush_m0	PVBus	Master	The core will generate bus requests on this port.
reset[1]	Signal	Slave	Raising this signal will put the core into reset mode.
standbywfe[1]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[1]	Signal	Master	This signal indicates if a core is in WFI state.
teinit[1]	Signal	Slave	This signal provides default exception handling state.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vinithi[1]	Signal	Slave	This signal controls the location of the exception vectors at reset.

Parameters for ARMCortexA9UPCT

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Initialize to BE8 endianness

Type

bool

Default value

0x0

cpuX.CFGNMFI

Enable nonmaskable FIQ interrupts on startup

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.POWERCTLI

Default power control state for processor

Type

int

Default value

0x0

cpuX.TEINIT

T32 exception enable. The default has exceptions including reset handled in A32 state

Type

bool

Default value

0x0

cpuX.VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpuX.ase-present

Set whether model has NEON support

Type

bool

Default value

0x1

cpuX.dcache-size

Set D-cache size in bytes

Type

int

Default value

0x8000

cpuX.icache-size

Set I-cache size in bytes

Type

int

Default value

0x8000

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

cpuX.semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

cpuX.semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

cpuX.vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

D-cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

D-cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

D-cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_access_latency

D-cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

D-cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

D-cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

D-cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

device-accurate-tlb

Specify whether all TLBs are modeled

Type

bool

Default value

0x0

icache-hit_latency

I-cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

I-cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

I-cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_access_latency

I-cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

I-cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

3.5.10 **ARMCortexA15x1CT**

ARMCortexA15x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-210: IP revisions support

Revision	Quality level
r2p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA15x1CT

This model has the following Iris instances:

Table 3-211: ARMCortexA15x1CT Iris instances

InstanceName	ComponentName
ARMCortexA15x1CT	Cluster_ARM_Cortex-A15
ARMCortexA15x1CT.Cortex-A15_GIC	GICv2
ARMCortexA15x1CT.acp_mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA15x1CT.cpu0	ARM_Cortex-A15
ARMCortexA15x1CT.cpu0.DTLB	TLB
ARMCortexA15x1CT.cpu0.ITLB	TLB
ARMCortexA15x1CT.cpu0.dtlb	TlbCadi
ARMCortexA15x1CT.cpu0.itlb	TlbCadi
ARMCortexA15x1CT.cpu0.l1dcache	PVCache
ARMCortexA15x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA15x1CT.cpu0.l1icache	PVCache
ARMCortexA15x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA15x1CT.ext_bus	PVBusLogger
ARMCortexA15x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA15x1CT.l2_cache	PVCache
ARMCortexA15x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA15x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-212: ARMCortexA15x1CT MTI instances

InstanceName	ComponentName
ARMCortexA15x1CT.Cortex-A15_GIC	GICv2
ARMCortexA15x1CT.acp_mapper	PVBusMapper
ARMCortexA15x1CT.cpu0	ARM_Cortex-A15
ARMCortexA15x1CT.cpu0.DTLB	TLB
ARMCortexA15x1CT.cpu0.ITLB	TLB

InstanceName	ComponentName
ARMCortexA15x1CT.cpu0.l1dcache	PVCache
ARMCortexA15x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA15x1CT.cpu0.l1icache	PVCache
ARMCortexA15x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA15x1CT.ext_bus	PVBusLogger
ARMCortexA15x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA15x1CT.l2_cache	PVCache
ARMCortexA15x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA15x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA15x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA15x1CT contains the following CADI targets:

- ARM_Cortex-A15
- Cluster_ARM_Cortex-A15
- PVCache
- TlbCadi

About ARMCortexA15x1CT

- The following components also exist:
 - ARMCortexA15x2CT.
 - ARMCortexA15x3CT.
 - ARMCortexA15x4CT.

The per-core parameters are preceded by `cpu n .`, where n identifies the core (0-3).

- The `ase-present` and `vfp-present` parameters configure the synthesis options:

vfp present and ase present

Neon™ and VFPv4-D32 are supported.

vfp present and ase not present

VFPv4-D16 is supported.

vfp not present and ase present

Illegal. Forces `vfp-present` to `true` so model has Neon and VFPv4-D32 support.

vfp not present and ase not present

Model has neither Neon nor VFPv4-D32 support.

- If you are using the `ARMCortexA15x_nCT` component on a VE model platform, the `PERIPHBASE` parameter is set automatically to `0x2C000000` and is not visible in the parameter list.

ACE limitation

AXI Coherency Extensions (ACE) are extensions to AXI4 that support system-level cache coherency between multiple clusters. The ACE cache models in the Arm® Cortex®-A15 and Cortex-A7, and the ACE support in the CCI-400 have the limitation that they only process one transaction at a time. Normally, the simulation processes each transaction to completion before allowing any master to generate another transaction. However, in the following situation the simulation might fail. If a SystemC bus slave calls `wait()` while it is processing a transaction, this call might allow another master to issue another transaction that passes through the CCI-400 or the Cortex-A15 or Cortex-A7 caches. This situation could happen if a SystemC bus master running in another thread is connected to one of the ACE-lite ports on the CCI-400.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- The GIC does not respect the `CFGSDISABLE` signal. This leads to some registers wrongly being accessible.
- The Broadcast *Translation Lookaside Buffer* (TLB) or cache operations in the model do not cause other cores in the cluster that are asleep because of *Wait For Interrupt* (WFI) to wake up.
- It ignores the RR bit in the SCTLR.
- It implements the Power Control Register in the system control coprocessor but writing to it does not change the behavior of the model.
- When modeling the SCU, coherency operations are by memory writes then reads to refill from memory, rather than cache-to-cache transfers.
- It does not implement ETM registers.
- It implements TLB bitmap registers as RAZ/WI.
- It does not support the Cortex®-A15 mechanism to read the internal memory that the Cache and TLB structures use through the implementation defined region of the system coprocessor interface. This includes the RAM Index Register, IL1DATA Registers, DL1DATA Registers, and associated functionality.

- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.
- ECC and parity schemes are hardware-specific so are not supported.

Ports for ARMCortexA15x1CT

Table 3-213: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>broadcastcachemaint</code>	Signal	Slave	Enable broadcasting of cache maintenance operations to downstream caches.
<code>broadcastinner</code>	Signal	Slave	Enable broadcasting of Inner Shareable transactions.
<code>broadcastouter</code>	Signal	Slave	Enable broadcasting of Outer Shareable transactions.
<code>cfgend[4]</code>	Signal	Slave	This signal controls the SCTLR.EE bit.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>clusterid</code>	Value	Slave	This port sets the value in the CLUSTERID field (bits[11:8]) of the MPIDR.
<code>CNTHPIRQ[4]</code>	Signal	Master	Outputs of the generic timers.
<code>CNTPNSIRQ[4]</code>	Signal	Master	Outputs of the generic timers.
<code>CNTPSIRQ[4]</code>	Signal	Master	Outputs of the generic timers.
<code>cntvalueb</code>	CounterInterface	Slave	Interface to SoC level counter module.
<code>CNTVIRQ[4]</code>	Signal	Master	Outputs of the generic timers.
<code>cp15sdisable[4]</code>	Signal	Slave	This signal disables write access to some secure system control processor registers.
<code>cpuporeset[4]</code>	Signal	Slave	Signal initializes all processor logic including NEON, VFP, Debug, PTM, breakpoint and watchpoint.
<code>event</code>	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
<code>fiq[4]</code>	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
<code>fiqout[4]</code>	Signal	Master	This signal exports internal VGIC FIQ signal to the CPU..
<code>irq[4]</code>	Signal	Slave	This signal drives the CPU's interrupt handling.
<code>irqout[4]</code>	Signal	Master	This signal exports internal VGIC IRQ signal to the CPU.
<code>irqs[224]</code>	Signal	Slave	These signals drive the CPU's interrupt controller interrupt lines.
<code>l2reset</code>	Signal	Slave	This signal resets the shared L2 memory system, interrupt controller and timer logic.
<code>periphbase</code>	Value_64	Slave	This port sets the base address of the private peripheral region.
<code>pmuirq[4]</code>	Signal	Master	Interrupt signal from performance monitoring unit.
<code>presetdbg</code>	Signal	Slave	Signal initializes the shared Debug APB, CTI and CTM logic.
<code>pvbus_m0</code>	PVBus	Master	The core will generate bus requests on this port.
<code>reset[4]</code>	Signal	Slave	Raising this signal will put the core into reset mode.
<code>standbywfe[4]</code>	Signal	Master	This signal indicates if a core is in WFE state.
<code>standbywfi[4]</code>	Signal	Master	This signal indicates if a core is in WFI state.
<code>standbywfil2</code>	Signal	Master	Indicate that all the individual processors and the L2 memory system are in a WFI state.

Name	Protocol	Type	Description
teinit[4]	Signal	Slave	This signal enables Thumb exceptions (controls the SCTLR.TE bit).
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vfiq[4]	Signal	Slave	Virtual FIQ inputs. Note that the fiq pins are wired directly to the core if there is no internal VGIC. If there is an internal VGIC then these are ignored.
vinithi[4]	Signal	Slave	This signal controls the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtual IRQ inputs. Note that the irq pins are wired directly to the core if there is no internal VGIC. If there is an internal VGIC then these are ignored.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).

Parameters for ARM Cortex A15x1CT

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CFGSDISABLE

Disable some accesses to GIC registers

Type

bool

Default value

0x0

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

IMINLN

Instruction cache minimum line size: false=32 bytes, true=64 bytes

Type

bool

Default value

0x1

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Initialize to BE8 endianness

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.DBGROMADDR

This value is used to initialize the CP15 DBGDRAR register. Bits[39:12] of this register specify the ROM table physical address

Type

int

Default value

0x12000003

cpuX.DBGROMADDRV

If true, this sets bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x1

cpuX.DBGSELFADDR

This value is used to initialize the CP15 DBGDSAR register. Bits[39:17] of this register specify the ROM table physical address

Type

int

Default value

0x10003

cpuX.DBGSELFADDRV

If true, this sets bits[1:0] of the CP15 DBGDSAR to indicate that the address is valid

Type

bool

Default value

0x1

cpuX.TEINIT

T32 exception enable. The default has exceptions including reset handled in A32 state

Type

bool

Default value

0x0

cpuX.VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpuX.ase-present

Set whether CT model has been built with NEON support

Type

bool

Default value

0x1

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

cpuX.semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

cpuX.semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

cpuX.vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether CT model has been built with VFP support

Type

bool

Default value

0x1

dic-spi_count

Number of shared peripheral interrupts implemented

Type

int

Default value

0x40

disable_periph_decoder

Disable memory mapped access to gic system registers

Type

bool

Default value

0x0

internal_vgic

Configures whether the model of the processor contains a Virtualized Generic Interrupt Controller (VGIC)

Type

bool

Default value

0x1

l1_dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. l1_dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-state_modelled

Set whether L1 D-cache has stateful implementation

Type

bool

Default value

0x0

l1_dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. l1_dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. l1_icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-state_modelled

Set whether L1 I-cache has stateful implementation

Type

bool

Default value

0x0

l2-data-slice

L2 data RAM slice

Type

int

Default value

0x0

l2-tag-slice

L2 tag RAM slice

Type

int

Default value

0x0

l2_cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2_cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2_cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-size

Set L2 cache size in bytes

Type

int

Default value

0x80000

l2_cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2_cache-state_modelled

Set whether L2 cache has stateful implementation

Type

bool

Default value

0x0

l2_cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2_cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2_cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

3.5.11 ARMCortexA17x1CT

ARMCortexA17x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-214: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA17x1CT

This model has the following Iris instances:

Table 3-215: ARMCortexA17x1CT Iris instances

InstanceName	ComponentName
ARMCortexA17x1CT	Cluster_ARM_Cortex-A17
ARMCortexA17x1CT.acp_mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA17x1CT.cpu0	ARM_Cortex-A17
ARMCortexA17x1CT.cpu0.DTLB	TLB
ARMCortexA17x1CT.cpu0.ITLB	TLB
ARMCortexA17x1CT.cpu0.dtlb	TlbCadi
ARMCortexA17x1CT.cpu0.itlb	TlbCadi
ARMCortexA17x1CT.cpu0.l1dcache	PVCache
ARMCortexA17x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA17x1CT.cpu0.l1licache	PVCache
ARMCortexA17x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA17x1CT.ext_bus	PVBusLogger
ARMCortexA17x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA17x1CT.l2_cache	PVCache
ARMCortexA17x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA17x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-216: ARMCortexA17x1CT MTI instances

InstanceName	ComponentName
ARMCortexA17x1CT.acp_mapper	PVBusMapper
ARMCortexA17x1CT.cpu0	ARM_Cortex-A17
ARMCortexA17x1CT.cpu0.DTLB	TLB
ARMCortexA17x1CT.cpu0.ITLB	TLB
ARMCortexA17x1CT.cpu0.l1dcache	PVCache

InstanceName	ComponentName
ARMCortexA17x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA17x1CT.cpu0.l1icache	PVCache
ARMCortexA17x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA17x1CT.ext_bus	PVBusLogger
ARMCortexA17x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA17x1CT.l2_cache	PVCache
ARMCortexA17x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA17x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA17x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA17x1CT contains the following CADI targets:

- ARM_Cortex-A17
- Cluster_ARM_Cortex-A17
- PVCache
- TlbCadi

About ARMCortexA17x1CT

- The following components also exist:
 - ARMCortexA17x2CT.
 - ARMCortexA17x3CT.
 - ARMCortexA17x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

- The `ase-present` and `vfp-present` parameters configure the synthesis options:

vfp present and ase present

NEON and VFPv4-D32 are supported.

vfp present and ase not present

VFPv4-D16 is supported.

vfp not present and ase present

Illegal. Forces `vfp-present` to `true` so model has NEON and VFPv4-D32 support.

vfp not present and ase not present

Model has neither NEON nor VFPv4-D32 support.

- This model exposes the `BROADCASTCACHEMAINT`, `BROADCASTINNER`, and `BROADCASTOUTER` parameters at the CPU level, rather than at the cluster level. To achieve correct behavior, set the same value for all CPUs in the cluster.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.
- ECC support is hardware-specific so is not modeled.

Ports for ARMCortexA17x1CT**Table 3-217: Ports**

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>broadcastcachemaint</code>	Signal	Slave	Enable broadcasting of cache maintenance operations to downstream caches.
<code>broadcastinner</code>	Signal	Slave	Enable broadcasting of Inner Shareable transactions.
<code>broadcastouter</code>	Signal	Slave	Enable broadcasting of Outer Shareable transactions.
<code>CFGADDRFILTEENDNS</code>	Value_64	Slave	NS end address (only bits 39:20 are used).
<code>CFGADDRFILTEENDS</code>	Value_64	Slave	S end address (only bits 39:20 are used).
<code>CFGADDRFILTEENNS</code>	Signal	Slave	Enable periph port filtering for NS accesses.
<code>CFGADDRFILTEENS</code>	Signal	Slave	Enable periph port filtering for S accesses.
<code>CFGADDRFILTESTARTNS</code>	Value_64	Slave	NS start address (only bits 39:20 are used).
<code>CFGADDRFILTESTARTS</code>	Value_64	Slave	S start address (only bits 39:20 are used).
<code>cfgend[4]</code>	Signal	Slave	This signal controls the SCTL.R.EE bit.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>clusterid</code>	Value	Slave	This port sets the value in the CLUSTERID field (bits[11:8]) of the MPIDR.
<code>CNTHPIRQ[4]</code>	Signal	Master	Outputs of the generic timers.
<code>CNTPNSIRQ[4]</code>	Signal	Master	Outputs of the generic timers.
<code>CNTPSIRQ[4]</code>	Signal	Master	Outputs of the generic timers.
<code>cntvalueb</code>	CounterInterface	Slave	Interface to SoC level counter module.
<code>CNTVIRQ[4]</code>	Signal	Master	Outputs of the generic timers.
<code>cp15sdisable[4]</code>	Signal	Slave	This signal disables write access to some secure system control processor registers.
<code>cpuporeset[4]</code>	Signal	Slave	Signal initializes all processor logic including NEON, VFP, Debug, PTM, breakpoint and watchpoint.

Name	Protocol	Type	Description
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
irq[4]	Signal	Slave	This signal drives the CPU's interrupt handling.
l2flushdone	Signal	Master	Flush of L2 memory system complete.
l2flushreq	Signal	Slave	Request flush of L2 memory system.
l2reset	Signal	Slave	This signal resets the shared L2 memory system and timer logic.
peripheral_m	PVBus	Master	The core's peripheral port. Controlled by filter registers.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Signal initializes the shared Debug APB, CTI and CTM logic.
pvbush_m0	PVBus	Master	The core will generate bus requests on this port.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state.
standbywfil2	Signal	Master	Indicate that all the individual processors and the L2 memory system are in a WFI state.
teinit[4]	Signal	Slave	This signal enables Thumb exceptions (controls the SCTLR.TE bit).
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vfiq[4]	Signal	Slave	Virtual FIQ inputs.
vinithi[4]	Signal	Slave	This signal controls the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtual IRQ inputs.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).

Parameters for ARMCortexA17x1CT

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CFGADDRFILTEENDNS

Peripheral port NS end address

Type

int

Default value

0x0

CFGADDRFILTEENDS

Peripheral port S end address

Type

int

Default value

0x0

CFGADDRFILTEENNS

Peripheral port NS address filtering enabled

Type

bool

Default value

0x0

CFGADDRFILTEENS

Peripheral port S address filtering enabled

Type

bool

Default value

0x0

CFGADDRFILTSTARTNS

Peripheral port NS start address

Type

int

Default value

0x0

CFGADDRFILTSTARTS

Peripheral port S start address

Type

int

Default value

0x0

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

IMINLN

Instruction cache minimum line size: false=32 bytes, true=64 bytes

Type

bool

Default value

0x1

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Initialize to BE8 endianness

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.DBGROMADDR

This value is used to initialize the CP15 DBGDRAR register. Bits[39:12] of this register specify the ROM table physical address

Type

int

Default value

0x12000003

cpuX.DBGROMADDRV

If true, this sets bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x1

cpuX.DBGSELFADDR

This value is used to initialize the CP15 DBGDSAR register. Bits[39:17] of this register specify the ROM table physical address

Type

int

Default value

0x10003

cpuX.DBGSELFADDRV

If true, this sets bits[1:0] of the CP15 DBGDSAR to indicate that the address is valid

Type

bool

Default value

0x1

cpuX.TEINIT

T32 exception enable. The default has exceptions including reset handled in A32 state

Type

bool

Default value

0x0

cpuX.VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpuX.ase-present

Set whether CT model has been built with NEON support

Type

bool

Default value

0x1

cpuX.l1_icode-size

Size of L1 I-cache

Type

int

Default value

0x8000

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

cpuX.semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

cpuX.semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

cpuX.vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether CT model has been built with VFP support

Type

bool

Default value

0x1

l1_dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. l1_dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-state_modelled

Set whether L1 D-cache has stateful implementation

Type

bool

Default value

0x0

l1_dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. l1_dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l1_icache-

read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. l1_icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l1_icache-state_modelled

Set whether L1 I-cache has stateful implementation

Type

bool

Default value

0x0

l2-data-slice

L2 data RAM slice

Type

int

Default value

0x0

l2-tag-slice

L2 tag RAM slice

Type

int

Default value

0x0

l2_cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2_cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2_cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-size

Set L2 cache size in bytes

Type

int

Default value

0x40000

l2_cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-state_modelled

Set whether L2 cache has stateful implementation

Type

bool

Default value

0x0

l2_cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2_cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2_cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed.

l2_cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

3.5.12 ARMCortexA32x1CT

ARMCortexA32x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-218: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA32x1CT

This model has the following Iris instances:

Table 3-219: ARMCortexA32x1CT Iris instances

InstanceName	ComponentName
ARMCortexA32x1CT	Cluster_ARM_Cortex-A32
ARMCortexA32x1CT.AMU	PVBusLogger
ARMCortexA32x1CT.AMU.mapper	PVBusMapper
ARMCortexA32x1CT.DAP	PVBusLogger
ARMCortexA32x1CT.DAP.mapper	PVBusMapper
ARMCortexA32x1CT.DSU	DSU
ARMCortexA32x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA32x1CT.MMAP	PVBusLogger
ARMCortexA32x1CT.MMAP.mapper	PVBusMapper
ARMCortexA32x1CT.acp_mapper	PVBusMapper
ARMCortexA32x1CT.cpu0	ARM_Cortex-A32
ARMCortexA32x1CT.cpu0.S1TLB	TLB
ARMCortexA32x1CT.cpu0.S2TLB	TLB
ARMCortexA32x1CT.cpu0.UTLB	TLB
ARMCortexA32x1CT.cpu0.dtlb	TlbCadi
ARMCortexA32x1CT.cpu0.l1dcache	PVCache

InstanceName	ComponentName
ARMCortexA32x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA32x1CT.cpu0.l1icache	PVCache
ARMCortexA32x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA32x1CT.ext_bus	PVBusLogger
ARMCortexA32x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA32x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA32x1CT.l2_cache	PVCache
ARMCortexA32x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA32x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-220: ARMCortexA32x1CT MTI instances

InstanceName	ComponentName
ARMCortexA32x1CT	ARMv8Cluster
ARMCortexA32x1CT.AMU	PVBusLogger
ARMCortexA32x1CT.AMU.mapper	PVBusMapper
ARMCortexA32x1CT.DAP	PVBusLogger
ARMCortexA32x1CT.DAP.mapper	PVBusMapper
ARMCortexA32x1CT.DSU	DSU
ARMCortexA32x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA32x1CT.MMAP	PVBusLogger
ARMCortexA32x1CT.MMAP.mapper	PVBusMapper
ARMCortexA32x1CT.acp_mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA32x1CT.cpu0	ARM_Cortex-A32
ARMCortexA32x1CT.cpu0.S1TLB	TLB
ARMCortexA32x1CT.cpu0.S2TLB	TLB
ARMCortexA32x1CT.cpu0.UTLB	TLB
ARMCortexA32x1CT.cpu0.l1dcache	PVCache
ARMCortexA32x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA32x1CT.cpu0.l1icache	PVCache
ARMCortexA32x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA32x1CT.ext_bus	PVBusLogger
ARMCortexA32x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA32x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA32x1CT.l2_cache	PVCache
ARMCortexA32x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA32x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA32x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA32x1CT contains the following CADI targets:

- ARM_Cortex-A32
- Cluster_ARM_Cortex-A32
- PVCache
- TlbCadi

About ARMCortexA32x1CT

- The following components also exist:

- ARMCortexA32x2CT.
- ARMCortexA32x3CT.
- ARMCortexA32x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

- If either L1 cache is stateful, then the L2 cache is stateful. This is controlled by the `dcache-state_modelled` and `icache-state_modelled` parameters.
- The cache latency parameters are only effective when you enable cache-state modeling.
- Timing annotation for transactions downstream of the cache and TLB models propagates through the models.
- Although Neon™ support is optional for the Arm® Cortex®-A32 processor, this model does not implement the `ase-present` parameter. Therefore, it is not possible to configure the model to not support Neon.
- This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).
- The `semihosting-cwd` parameter sets the CWD that is used for semihosting. The host operating system limits the maximum path length. The `semihosting-cwd` parameter does not provide any security. Software running on the model can access files outside this directory using relative paths containing `..` or using absolute paths.
- The system designer decides whether a debug APB ties the external debug view with other system views. In the model, use a `PVBusDecoder` to direct traffic to the correct port, `dev_debug_s` or `memorymapped_debug_s`.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.
- ECC and parity schemes are hardware-specific so are not supported.

Ports for ARMCortexA32x1CT

Table 3-221: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastinner</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>cfgend[4]</code>	Signal	Slave	This signal if for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>cfgte[4]</code>	Signal	Slave	This signal provides default exception handling state.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.

Name	Protocol	Type	Description
clrexmonack	Signal	Master	Acknowledge handshake signal for the clrexmonreq signal
clrexmonreq	Signal	Slave	Signals the clearing of an external global exclusive monitor
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
CNTHPIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Timer signals to SOC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
commrx[4]	Signal	Master	Receive portion of Data Transfer Register full.
commtx[4]	Signal	Master	Transmit portion of Data Transfer Register empty.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers.
cp15sdisable2[4]	Signal	Slave	-
cpuporeset[4]	Signal	Slave	CPU power on reset.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port
ctidbgirq[4]	Signal	Master	Cross trigger matrix port.
dbgack[4]	Signal	Master	External debug interface.
dbgen[4]	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Processor powerup request.
dev_debug_s	PVBus	Slave	External debug interface.
edbgrq[4]	Signal	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
l2flushdone	Signal	Master	Flush of L2 memory system complete.
l2flushreq	Signal	Slave	Request flush of L2 memory system.
l2reset	Signal	Slave	Level2 reset.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[4]	Signal	Slave	External debug interface.

Name	Protocol	Type	Description
periphbase	Value_64	Slave	This port sets the base address of private peripheral region.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Debug reset.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt
reset[4]	Signal	Slave	Reset.
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
sei[4]	Signal	Slave	Per core System Error physical pins.
smpen[4]	Signal	Master	This signals AMP or SMP mode for each core.
spiden[4]	Signal	Slave	External debug interface.
spniden[4]	Signal	Slave	External debug interface.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state
standbywfi2	Signal	Master	This signal indicated all cores and L2 are idles and in low power state.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA32x1CT

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x22000000

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CP15SDISABLE2

Initialize to disable access to some CP15 registers (FEAT_CP15SDISABLE2).

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit

or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`dcache-read_latency`

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. `dcache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`dcache-size`

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

`dcache-snoop_data_transfer_latency`

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`dcache-state_modelled`

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.13 ARMCortexA34x1CT

ARMCortexA34x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-222: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA34x1CT

This model has the following Iris instances:

Table 3-223: ARMCortexA34x1CT Iris instances

InstanceName	ComponentName
ARMCortexA34x1CT	Cluster_ARM_Cortex-A34
ARMCortexA34x1CT.AMU	PVBusLogger
ARMCortexA34x1CT.AMU.mapper	PVBusMapper
ARMCortexA34x1CT.DAP	PVBusLogger
ARMCortexA34x1CT.DAP.mapper	PVBusMapper
ARMCortexA34x1CT.DSU	DSU
ARMCortexA34x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA34x1CT.MMAP	PVBusLogger
ARMCortexA34x1CT.MMAP.mapper	PVBusMapper
ARMCortexA34x1CT.acp_mapper	PVBusMapper
ARMCortexA34x1CT.cpu0	ARM_Cortex-A34
ARMCortexA34x1CT.cpu0.UTLB	TLB
ARMCortexA34x1CT.cpu0.dtlb	TlbCadi
ARMCortexA34x1CT.cpu0.l1dcache	PVCache
ARMCortexA34x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA34x1CT.cpu0.l1licache	PVCache
ARMCortexA34x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA34x1CT.ext_bus	PVBusLogger
ARMCortexA34x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA34x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA34x1CT.l2_cache	PVCache
ARMCortexA34x1CT.l2_cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA34x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA34x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-224: ARMCortexA34x1CT MTI instances

InstanceName	ComponentName
ARMCortexA34x1CT	ARMv8Cluster
ARMCortexA34x1CT.AMU	PVBusLogger
ARMCortexA34x1CT.AMU.mapper	PVBusMapper
ARMCortexA34x1CT.DAP	PVBusLogger
ARMCortexA34x1CT.DAP.mapper	PVBusMapper
ARMCortexA34x1CT.DSU	DSU
ARMCortexA34x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA34x1CT.MMAP	PVBusLogger
ARMCortexA34x1CT.MMAP.mapper	PVBusMapper
ARMCortexA34x1CT.acp_mapper	PVBusMapper
ARMCortexA34x1CT.cpu0	ARM_Cortex-A34
ARMCortexA34x1CT.cpu0.UTLB	TLB
ARMCortexA34x1CT.cpu0.l1dcache	PVCache
ARMCortexA34x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA34x1CT.cpu0.l1icache	PVCache
ARMCortexA34x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA34x1CT.ext_bus	PVBusLogger
ARMCortexA34x1CT.ext_bus.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA34x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA34x1CT.l2_cache	PVCache
ARMCortexA34x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA34x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA34x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA34x1CT contains the following CADI targets:

- ARM_Cortex-A34
- Cluster_ARM_Cortex-A34
- PVCache
- TlbCadi

About ARMCortexA34x1CT

The model has the following features:

- If either L1 cache is stateful, then the L2 cache is stateful. This is controlled by the `dcache-state_modelled` and `icache-state_modelled` parameters.
- The cache latency parameters are only effective when you enable cache-state modeling.
- Timing annotation for transactions downstream of the cache and TLB models propagates through the models.
- Although Neon™ support is optional for this processor, this model does not implement the `ase-present` parameter. Therefore, it is not possible to configure the model to not support Neon.
- This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

- The `semihosting-cwd` parameter sets the CWD that is used for semihosting. The host operating system limits the maximum path length. The `semihosting-cwd` parameter does not provide any security. Software running on the model can access files outside this directory using relative paths containing `..` or using absolute paths.
- The system designer decides whether a debug APB ties the external debug view with other system views. In the model, use a `PVBusDecoder` to direct traffic to the correct port, `dev_debug_s` or `memorymapped_debug_s`.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.
- ECC and parity schemes are hardware-specific so are not supported.
- The following components also exist:
 - `ARMCortexA34x2CT`
 - `ARMCortexA34x3CT`
 - `ARMCortexA34x4CT`

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

Ports for ARMCortexA34x1CT

Table 3-225: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcasttinner</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>cfgend[4]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>cfgte[4]</code>	Signal	Slave	This signal provides default exception handling state.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>clrexmonack</code>	Signal	Master	Acknowledge handshake signal for the <code>clrexmonreq</code> signal
<code>clrexmonreq</code>	Signal	Slave	Signals the clearing of an external global exclusive monitor
<code>clusterid</code>	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
<code>CNTHPIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>CNTPNSIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>CNTPSIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>cntvalueb</code>	CounterInterface	Slave	Interface to SoC level counter module.
<code>CNTVIRQ[4]</code>	Signal	Master	Timer signals to SOC.

Name	Protocol	Type	Description
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
commrx[4]	Signal	Master	Receive portion of Data Transfer Register full.
commtx[4]	Signal	Master	Transmit portion of Data Transfer Register empty.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[4]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgack[4]	Signal	Master	External debug interface.
dbgen[4]	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Processor powerup request.
dev_debug_s	PVBus	Slave	External debug interface.
edbgrq[4]	Signal	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
l2flushdone	Signal	Master	Flush of L2 memory system complete.
l2flushreq	Signal	Slave	Request flush of L2 memory system.
l2reset	Signal	Slave	This signal resets the shared L2 memory system, interrupt controller and timer logic.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[4]	Signal	Slave	External debug interface.
periphbase	Value_64	Slave	This port sets the base address of private peripheral region.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt.

Name	Protocol	Type	Description
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sei[4]	Signal	Slave	Per core virtual System Error physical pins.
smpen[4]	Signal	Master	This signals AMP or SMP mode for each core.
spiden[4]	Signal	Slave	External debug interface.
spniden[4]	Signal	Slave	External debug interface.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state.
standbywfi12	Signal	Master	Indicate that all the individual processors and the L2 memory system are in a WFI state.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A34x1CT

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CP15SDISABLE2

Initialize to disable access to some CP15 registers (FEAT_CP15SDISABLE2).

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.14 ARMCortexA35x1CT

ARMCortexA35x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-226: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.**Iris and MTI instances for ARMCortexA35x1CT**

This model has the following Iris instances:

Table 3-227: ARMCortexA35x1CT Iris instances

InstanceName	ComponentName
ARMCortexA35x1CT	Cluster_ARM_Cortex-A35
ARMCortexA35x1CT.AMU	PVBusLogger
ARMCortexA35x1CT.AMU.mapper	PVBusMapper
ARMCortexA35x1CT.DAP	PVBusLogger
ARMCortexA35x1CT.DAP.mapper	PVBusMapper
ARMCortexA35x1CT.DSU	DSU
ARMCortexA35x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA35x1CT.MMAP	PVBusLogger
ARMCortexA35x1CT.MMAP.mapper	PVBusMapper
ARMCortexA35x1CT.acp_mapper	PVBusMapper
ARMCortexA35x1CT.cpu0	ARM_Cortex-A35
ARMCortexA35x1CT.cpu0.UTLB	TLB
ARMCortexA35x1CT.cpu0.dtlb	TlbCadi
ARMCortexA35x1CT.cpu0.l1dcache	PVCache
ARMCortexA35x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA35x1CT.cpu0.l1licache	PVCache
ARMCortexA35x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA35x1CT.ext_bus	PVBusLogger
ARMCortexA35x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA35x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA35x1CT.l2_cache	PVCache
ARMCortexA35x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA35x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-228: ARMCortexA35x1CT MTI instances

InstanceName	ComponentName
ARMCortexA35x1CT	ARMv8Cluster
ARMCortexA35x1CT.AMU	PVBusLogger
ARMCortexA35x1CT.AMU.mapper	PVBusMapper
ARMCortexA35x1CT.DAP	PVBusLogger
ARMCortexA35x1CT.DAP.mapper	PVBusMapper
ARMCortexA35x1CT.DSU	DSU
ARMCortexA35x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA35x1CT.MMAP	PVBusLogger
ARMCortexA35x1CT.MMAP.mapper	PVBusMapper
ARMCortexA35x1CT.acp_mapper	PVBusMapper
ARMCortexA35x1CT.cpu0	ARM_Cortex-A35

InstanceName	ComponentName
ARMCortexA35x1CT.cpu0.UTLB	TLB
ARMCortexA35x1CT.cpu0.l1dcache	PVCache
ARMCortexA35x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA35x1CT.cpu0.l1icache	PVCache
ARMCortexA35x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA35x1CT.ext_bus	PVBusLogger
ARMCortexA35x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA35x1CT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder
ARMCortexA35x1CT.l2_cache	PVCache
ARMCortexA35x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA35x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA35x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA35x1CT contains the following CADI targets:

- ARM_Cortex-A35
- Cluster_ARM_Cortex-A35
- PVCache
- TlbCadi

About ARMCortexA35x1CT

The model has the following features:

- If either L1 cache is stateful, then the L2 cache is stateful. This is controlled by the `dcache-state_modelled` and `icache-state_modelled` parameters.
- The cache latency parameters are only effective when you enable cache-state modeling.

- Timing annotation for transactions downstream of the cache and TLB models propagates through the models.
- Although Neon™ support is optional for this processor, this model does not implement the `ase-present` parameter. Therefore, it is not possible to configure the model to not support Neon.
- This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).
- The `semihosting-cwd` parameter sets the CWD that is used for semihosting. The host operating system limits the maximum path length. The `semihosting-cwd` parameter does not provide any security. Software running on the model can access files outside this directory using relative paths containing `..` or using absolute paths.
- The system designer decides whether a debug APB ties the external debug view with other system views. In the model, use a `PVBusDecoder` to direct traffic to the correct port, `dev_debug_s` or `memorymapped_debug_s`.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.
- ECC and parity schemes are hardware-specific so are not supported.
- The following components also exist:
 - `ARMCortexA35x2CT`.
 - `ARMCortexA35x3CT`.
 - `ARMCortexA35x4CT`.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

Ports for ARMCortexA35x1CT

Table 3-229: Ports

Name	Protocol	Type	Description
<code>aa64naa32[4]</code>	Signal	Slave	Register width after reset.
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastinner</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>cfgend[4]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>cfgte[4]</code>	Signal	Slave	This signal provides default exception handling state.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>clrexmonack</code>	Signal	Master	Acknowledge handshake signal for the <code>clrexmonreq</code> signal

Name	Protocol	Type	Description
clrexmonreq	Signal	Slave	Signals the clearing of an external global exclusive monitor
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
CNTHPIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Timer signals to SOC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
commrx[4]	Signal	Master	Receive portion of Data Transfer Register full.
commtx[4]	Signal	Master	Transmit portion of Data Transfer Register empty.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers.
cp15sdisable2[4]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[4]	Signal	Slave	CPU power on reset.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross trigger matrix port.
dbgack[4]	Signal	Master	External debug interface.
dbgen[4]	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Processor powerup request.
dev_debug_s	PVBus	Slave	External debug interface.
edbgrq[4]	Signal	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
l2flushdone	Signal	Master	Flush of L2 memory system complete.
l2flushreq	Signal	Slave	Request flush of L2 memory system.
l2reset	Signal	Slave	Level2 reset.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[4]	Signal	Slave	External debug interface.
periphbase	Value_64	Slave	This port sets the base address of private peripheral region.

Name	Protocol	Type	Description
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Debug reset.
pvbusr_m0	PVBus	Master	The core will generate bus requests on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt
reset[4]	Signal	Slave	Reset.
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sei[4]	Signal	Slave	Per core System Error physical pins.
smpen[4]	Signal	Master	This signals AMP or SMP mode for each core.
spiden[4]	Signal	Slave	External debug interface.
spniden[4]	Signal	Slave	External debug interface.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state
standbywfil2	Signal	Master	This signal indicated all cores and L2 are idles and in low power state.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM CortexA35x1CT

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CP15SDISABLE2

Initialize to disable access to some CP15 registers (FEAT_CP15SDISABLE2).

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.15 **ARMCortexA53x1CT**

ARMCortexA53x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-230: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA53x1CT

This model has the following Iris instances:

Table 3-231: ARMCortexA53x1CT Iris instances

InstanceName	ComponentName
ARMCortexA53x1CT	Cluster_ARM_Cortex-A53
ARMCortexA53x1CT.AMU	PVBusLogger
ARMCortexA53x1CT.AMU.mapper	PVBusMapper
ARMCortexA53x1CT.DAP	PVBusLogger
ARMCortexA53x1CT.DAP.mapper	PVBusMapper
ARMCortexA53x1CT.DSU	DSU
ARMCortexA53x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA53x1CT.MMAP	PVBusLogger
ARMCortexA53x1CT.MMAP.mapper	PVBusMapper
ARMCortexA53x1CT.acp_mapper	PVBusMapper
ARMCortexA53x1CT.cpu0	ARM_Cortex-A53
ARMCortexA53x1CT.cpu0.UTLB	TLB
ARMCortexA53x1CT.cpu0.dtlb	TlbCadi
ARMCortexA53x1CT.cpu0.l1dcache	PVCache
ARMCortexA53x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA53x1CT.cpu0.l1licache	PVCache
ARMCortexA53x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA53x1CT.ext_bus	PVBusLogger
ARMCortexA53x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA53x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA53x1CT.l2_cache	PVCache
ARMCortexA53x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[9]	PVBusSlave

InstanceName	ComponentName
ARMCortexA53x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-232: ARMCortexA53x1CT MTI instances

InstanceName	ComponentName
ARMCortexA53x1CT	ARMv8Cluster
ARMCortexA53x1CT.AMU	PVBusLogger
ARMCortexA53x1CT.AMU.mapper	PVBusMapper
ARMCortexA53x1CT.DAP	PVBusLogger
ARMCortexA53x1CT.DAP.mapper	PVBusMapper
ARMCortexA53x1CT.DSU	DSU
ARMCortexA53x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA53x1CT.MMAP	PVBusLogger
ARMCortexA53x1CT.MMAP.mapper	PVBusMapper
ARMCortexA53x1CT.acp_mapper	PVBusMapper
ARMCortexA53x1CT.cpu0	ARM_Cortex-A53
ARMCortexA53x1CT.cpu0.UTLB	TLB
ARMCortexA53x1CT.cpu0.l1dcache	PVCache
ARMCortexA53x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA53x1CT.cpu0.l1licache	PVCache
ARMCortexA53x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA53x1CT.ext_bus	PVBusLogger
ARMCortexA53x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA53x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA53x1CT.l2_cache	PVCache
ARMCortexA53x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[6]	PVBusSlave

InstanceName	ComponentName
ARMCortexA53x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA53x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA53x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA53x1CT contains the following CADI targets:

- ARM_Cortex-A53
- Cluster_ARM_Cortex-A53
- PVCache
- TlbCadi

About ARMCortexA53x1CT

- The following components also exist:
 - ARMCortexA53x2CT.
 - ARMCortexA53x3CT.
 - ARMCortexA53x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

- This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).
- The system designer decides whether a debug APB ties the external debug view with other system views. In the model, use a `PVBusDecoder` to direct traffic to the correct port, `dev_debug_s` or `memorymapped_debug_s`.

Differences between the CT model and RTL implementations

This component has the following differences from the corresponding revision of the RTL implementation:

- The value of the AArch64 PMCEID0_ELO register, and the AArch32 alias of this register, differs in the model from the TRM value. The model value reflects the model counters.
- The mechanisms for setting the affinity fields of the MPIDR. The RTL has two ports:
 - CLUSTERIDAFF1[7:0].
 - CLUSTERIDAFF2[7:0].

AFF1 sets the value of MPIDR bits[15:8] and AFF2 sets the value of MPIDR bits[23:16]. In contrast, the model has a single `CLUSTER_ID` port. This difference allows the setting of bits[23:8] of the MPIDR using bits[15:0] of the `CLUSTER_ID` value.

- The memory mapped debug registers have a view for cores and a view for external debug agents. In the model, these views require two PVBus ports. In hardware, the system designer decides how the implementation differentiates the views.

- In the model, a single `peer` event port combines the functionality of the `eventi` and `evento` signals in the RTL.
- The Generic Timers are Programmer's View (PV)-level abstractions: a model-specific protocol connects the `cntvalueb` port to the `MemoryMappedCounterModule`.
- The GIC CPU Interface is a PV-level abstraction: a model-specific protocol connects the GIC CPU Interface to the GIC Distributor.
- The Coresight™ *Cross Trigger Interface* (CTI) is a PV-level abstraction: the interface is a model specific one.
- The model has no mechanism to read the internal memory that the Cache and TLB structures use, through the implementation defined region of the system coprocessor interface. This memory includes the RAM Index Register, IL1DATA Registers, DL1DATA Registers, and associated functionality.
- The model does not implement:
 - ETM registers.
 - The PMUEVENT bus.
 - The WARMRESETREQ signal. However, the warm reset code sequence (see the section *Code sequence to request a Warm reset as a result of RMR_ELx.RR* in the [Arm Architecture Reference Manual for A-profile architecture](#)) makes the model simulate a warm reset of the core.
 - The PMUSNAPSHOTREQ and PMUSNAPSHOTACK signals.
 - The EXTERRIRQ and INTERRIRQ signals.
 - Processor dynamic-retention signals.
 - The SYSBARDISABLE signal.
 - The DBGPWRDUP, DBGPWRUPREQ, DBGNOPWRDWN, and DBGRSTREQ debug power management signals.
 - The RTL synthesis option to remove FP and ASE.
 - The RTL synthesis option for a Arm® Cortex®-A15 style debug memory map.
 - Although NEON support is optional for the Cortex-A53 processor, this model does not implement the `ase-present` parameter. This means it is not possible to configure the model to not support NEON.
 - ECC and parity schemes are hardware-specific so are not supported.

Ports for ARMCortexA53x1CT

Table 3-233: Ports

Name	Protocol	Type	Description
<code>aa64naa32[4]</code>	Signal	Slave	Register width after reset.
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>broadcastcachemaint</code>	Signal	Slave	Enable broadcasting of cache maintenance operations to downstream caches.
<code>broadcastinner</code>	Signal	Slave	Enable broadcasting of Inner Shareable transactions.

Name	Protocol	Type	Description
broadcastouter	Signal	Slave	Enable broadcasting of Outer Shareable transactions.
cfgend[4]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[4]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clrexmonack	Signal	Master	Acknowledge handshake signal for the clrexmonreq signal
clrexmonreq	Signal	Slave	Signals the clearing of an external global exclusive monitor
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
CNTHPIRQ[4]	Signal	Master	Timer signals to SoC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SoC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SoC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Timer signals to SoC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
commr[4]	Signal	Master	Receive portion of Data Transfer Register full.
commtx[4]	Signal	Master	Transmit portion of Data Transfer Register empty.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[4]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgack[4]	Signal	Master	External debug interface.
dbgen[4]	Signal	Slave	External debug interface.
dbgllrstdisable	Signal	Slave	Control ram clear on reset
dbgnopwrdown[4]	Signal	Master	This signals relate to core power down.
dbgpwrupreq[4]	Signal	Master	This signals relate to core power down.
dev_debug_s	PVBus	Slave	External debug interface.
edbgrq[4]	Signal	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.

Name	Protocol	Type	Description
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
l2flushdone	Signal	Master	Flush of L2 memory system complete.
l2flushreq	Signal	Slave	Request flush of L2 memory system.
l2reset	Signal	Slave	Reset the shared L2 memory system controller.
l2rstdisable	Signal	Slave	-
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[4]	Signal	Slave	External debug interface.
periphbase	Value_64	Slave	This port sets the base address of private peripheral region.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
rei[4]	Signal	Slave	Individual processor RAM Error Interrupt signal input.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sei[4]	Signal	Slave	Per core System Error physical pins.
smpen[4]	Signal	Master	This signals AMP or SMP mode for each core.
spiden[4]	Signal	Slave	External debug interface.
spniden[4]	Signal	Slave	External debug interface.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state.
standbywfi12	Signal	Master	Indicate that all the individual processors and the L2 systems are in a WFI state.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtual FIQ.

Name	Protocol	Type	Description
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtual IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Processor Virtual System Error Interrupt request.

Parameters for ARM Cortex A53x1CT

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size `dcache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

patch_level

Patch level of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Revision field in MIDR/MIDR_EL1. Corresponds to the patch number Y in rXpY.

Type

int

Default value

0x1

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.16 ARMCortexA55CT

ARMCortexA55CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-234: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA55CT

This model has the following Iris instances:

Table 3-235: ARMCortexA55CT Iris instances

InstanceName	ComponentName
ARMCortexA55CT	Cluster_ARM_Cortex-A55
ARMCortexA55CT.AMU	PVBusLogger
ARMCortexA55CT.AMU.mapper	PVBusMapper
ARMCortexA55CT.DAP	PVBusLogger
ARMCortexA55CT.DAP.mapper	PVBusMapper
ARMCortexA55CT.DSU	DSU
ARMCortexA55CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA55CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA55CT.DSU.shared_cache	PVCache
ARMCortexA55CT.DSU.shared_cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA55CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA55CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA55CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA55CT.MMAP	PVBusLogger
ARMCortexA55CT.MMAP.mapper	PVBusMapper
ARMCortexA55CT.cpu0	ARM_Cortex-A55
ARMCortexA55CT.cpu0.UTLB	TLB
ARMCortexA55CT.cpu0.dtlb	TlbCadi
ARMCortexA55CT.cpu0.l1dcache	PVCache
ARMCortexA55CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT.cpu0.l1icache	PVCache
ARMCortexA55CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA55CT.cpu0.l2cache	PVCache
ARMCortexA55CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT.ext_bus	PVBusLogger
ARMCortexA55CT.ext_bus.mapper	PVBusMapper
ARMCortexA55CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-236: ARMCortexA55CT MTI instances

InstanceName	ComponentName
ARMCortexA55CT	ARMv8Cluster
ARMCortexA55CT.AMU	PVBusLogger
ARMCortexA55CT.AMU.mapper	PVBusMapper
ARMCortexA55CT.DAP	PVBusLogger
ARMCortexA55CT.DAP.mapper	PVBusMapper
ARMCortexA55CT.DSU	DSU
ARMCortexA55CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA55CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA55CT.DSU.shared_cache	PVCache
ARMCortexA55CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA55CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA55CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA55CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA55CT.MMAP	PVBusLogger
ARMCortexA55CT.MMAP.mapper	PVBusMapper
ARMCortexA55CT.cpu0	ARM_Cortex-A55
ARMCortexA55CT.cpu0.UTLB	TLB

InstanceName	ComponentName
ARMCortexA55CT.cpu0.l1dcache	PVCache
ARMCortexA55CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT.cpu0.l1icache	PVCache
ARMCortexA55CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA55CT.cpu0.l2cache	PVCache
ARMCortexA55CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT.ext_bus	PVBusLogger
ARMCortexA55CT.ext_bus.mapper	PVBusMapper
ARMCortexA55CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA55CT contains the following CADI targets:

- ARM_Cortex-A55
- Cluster_ARM_Cortex-A55
- PVCache
- TlbCadi

About ARMCortexA55CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.

- Snoop filtering.
- Debug power management signals `DBGPWRDUP` and `DBGIRSTREQ` are not implemented, but `DBGPWRUPREQ` and `DBGNOPWRDWN` are implemented.
- Cache stashing capability.
- Dual ACE masters.

This model has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following models also exist, with a fixed number of cores per cluster:

- ARMCortexA55x1CT.
- ARMCortexA55x2CT.
- ARMCortexA55x3CT.
- ARMCortexA55x4CT.
- ARMCortexA55x8CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-7).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA55CT

Table 3-237: Ports

Name	Protocol	Type	Description
<code>aa64naa32[8]</code>	Signal	Slave	Register width after reset.
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AENDMP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[8]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>cfgte[8]</code>	Signal	Slave	This signal provides default exception handling state.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.

Name	Protocol	Type	Description
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[8]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[8]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[8]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[8]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[8]	Signal	Master	Timer signals to SOC.
commirq[8]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[8]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cp15sdisable[8]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[8]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[8]	Signal	Slave	Disable cryptography extensions after reset.
cti[8]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[8]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[8]	Signal	Master	No power-down request.
dbgpwrupreq[8]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[8]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[8]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[8]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[8]	PChannel	Slave	PChannels for cores
pmuirq[8]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.

Name	Protocol	Type	Description
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[8]	Signal	Slave	Per core RAM Error Interrupt.
reset[8]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[8]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[8]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[8]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[8]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[8]	Signal	Slave	Virtualised FIQ.
vinithi[8]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[8]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[8]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA55CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x40000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.17 ARMCortexA55CT_CortexA75CT

ARMCortexA55CT_CortexA75CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-238: IP revisions support

Revision	Quality level
CortexA55 r1p0	Full support
CortexA75 r3p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- subcluster0.tlbi_stall_enabled
- subcluster1.tlbi_stall_enabled

Iris and MTI instances for ARMCortexA55CT_CortexA75CT

This model has the following Iris instances:

Table 3-239: ARMCortexA55CT_CortexA75CT Iris instances

InstanceName	ComponentName
ARMCortexA55CT_CortexA75CT	Cluster_ARM_Cortex-A55_Cortex-A75
ARMCortexA55CT_CortexA75CT.AMU	PVBusLogger
ARMCortexA55CT_CortexA75CT.AMU.mapper	PVBusMapper
ARMCortexA55CT_CortexA75CT.DAP	PVBusLogger
ARMCortexA55CT_CortexA75CT.DAP.mapper	PVBusMapper
ARMCortexA55CT_CortexA75CT.DSU	DSU
ARMCortexA55CT_CortexA75CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA55CT_CortexA75CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache	PVCache
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA55CT_CortexA75CT.MMAP	PVBusLogger
ARMCortexA55CT_CortexA75CT.MMAP.mapper	PVBusMapper
ARMCortexA55CT_CortexA75CT.ext_bus	PVBusLogger
ARMCortexA55CT_CortexA75CT.ext_bus.mapper	PVBusMapper
ARMCortexA55CT_CortexA75CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA55CT_CortexA75CT.subcluster0	Subcluster_ARM_Cortex-A55
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0	ARM_Cortex-A55
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster1	Subcluster_ARM_Cortex-A75
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0	ARM_Cortex-A75
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster1.cpu1.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-240: ARMCortexA55CT_CortexA75CT MTI instances

InstanceName	ComponentName
ARMCortexA55CT_CortexA75CT	ARMv8Cluster
ARMCortexA55CT_CortexA75CT.AMU	PVBusLogger
ARMCortexA55CT_CortexA75CT.AMU.mapper	PVBusMapper
ARMCortexA55CT_CortexA75CT.DAP	PVBusLogger
ARMCortexA55CT_CortexA75CT.DAP.mapper	PVBusMapper
ARMCortexA55CT_CortexA75CT.DSU	DSU
ARMCortexA55CT_CortexA75CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA55CT_CortexA75CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache	PVCache
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA55CT_CortexA75CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA55CT_CortexA75CT.MMAP	PVBusLogger
ARMCortexA55CT_CortexA75CT.MMAP.mapper	PVBusMapper
ARMCortexA55CT_CortexA75CT.ext_bus	PVBusLogger
ARMCortexA55CT_CortexA75CT.ext_bus.mapper	PVBusMapper
ARMCortexA55CT_CortexA75CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0	ARM_Cortex-A55
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0	ARM_Cortex-A75
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.UTLB	TLB

InstanceName	ComponentName
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA75CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA55CT_CortexA75CT contains the following CADI targets:

- ARM_Cortex-A55
- ARM_Cortex-A75
- Cluster_ARM_Cortex-A55_Cortex-A75
- PVCache
- Subcluster_ARM_Cortex-A55
- Subcluster_ARM_Cortex-A75
- TlbCadi

About ARMCortexA55CT_CortexA75CT

The number of cores in each subcluster is configurable using the following parameters:

subcluster0.NUM_CORES

Possible values are 1-7 (ARMCortexA55CT).

subcluster1.NUM_CORES

Possible values are 1-4 (ARMCortexA75CT).

The total number of cores in the cluster cannot exceed 8.

Port arrays are expanded to the width needed for the maximum number of cores in each subcluster. Use the following port array indexes:

- `<port_name>[0-6]` for cores in subcluster0.
- `<port_name>[7-10]` for cores in subcluster1.



All instances in the Master cross trigger matrix port array, `cti[11]` must be connected, regardless of the `NUM_CORES` value used.

Core-specific parameters have the following prefixes:

- `subcluster0.cpu0` to `subcluster0.cpu6` identify cores in subcluster0.
- `subcluster1.cpu0` to `subcluster1.cpu3` identify cores in subcluster1.

For information about the cores in this model, see:

- [3.5.16 ARMCortexA55CT](#) on page 970.
- [3.5.26 ARMCortexA75CT](#) on page 1239.

See also [Arm DynamIQ Shared Unit Technical Reference Manual](#).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA55CT_CortexA75CT

Table 3-241: Ports

Name	Protocol	Type	Description
aa64naa32[11]	Signal	Slave	Register width after reset.
acp_s	PVBus	Slave	AXI ACP slave port
AENDMP	Value_64	Slave	DynamIQ port to obtain end address of valid peripheral address range (exclusive).
ASTARTMP	Value_64	Slave	DynamIQ port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[11]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[11]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamIQ pmu irq
CNTHPIRQ[11]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[11]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[11]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[11]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[11]	Signal	Master	Timer signals to SOC.
commirq[11]	Signal	Master	Interrupt signal from debug communications channel.

Name	Protocol	Type	Description
core_clk_in[11]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cp15sdisable[11]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[11]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[11]	Signal	Slave	Disable cryptography extensions after reset.
cti[11]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[11]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[11]	Signal	Master	No power-down request.
dbgprupreq[11]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[11]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[11]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[11]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[11]	PChannel	Slave	PChannels for cores
pmuirq[11]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[11]	Signal	Slave	Per core RAM Error Interrupt.
reset[11]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[11]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[11]	Signal	Slave	Per core System Error physical pins.

Name	Protocol	Type	Description
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[11]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[11]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[11]	Signal	Slave	Virtualised FIQ.
vinithi[11]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[11]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L3 system and becomes coherent (assuming attributes are set correctly).
vsei[11]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM CortexA55CT_CortexA75CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster0.cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.VINITHI

Reset value of SCTL.R.V.

Type

bool

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x40000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.CCSIDR-L1D_override

If nonzero, override the value presented in CCSIDR for L1D (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

subcluster1.CCSIDR-L1I_override

If nonzero, override the value presented in CCSIDR for L1I (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

subcluster1.CCSIDR-L2_override

If nonzero, override the value presented in CCSIDR for L2 (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster1.cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.18 ARM CortexA55CT_CortexA76CT

ARM CortexA55CT_CortexA76CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-242: IP revisions support

Revision	Quality level
CortexA55 r1p0	Full support
CortexA76 r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- subcluster0.tlbi_stall_enabled
- subcluster1.tlbi_stall_enabled

Iris and MTI instances for ARM CortexA55CT_CortexA76CT

This model has the following Iris instances:

Table 3-243: ARM CortexA55CT_CortexA76CT Iris instances

InstanceName	ComponentName
ARM CortexA55CT_CortexA76CT	Cluster_ARM_Cortex-A55_Cortex-A76
ARM CortexA55CT_CortexA76CT.AMU	PVBusLogger
ARM CortexA55CT_CortexA76CT.AMU.mapper	PVBusMapper
ARM CortexA55CT_CortexA76CT.DAP	PVBusLogger
ARM CortexA55CT_CortexA76CT.DAP.mapper	PVBusMapper
ARM CortexA55CT_CortexA76CT.DSU	DSU
ARM CortexA55CT_CortexA76CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM CortexA55CT_CortexA76CT.DSU.mpam_busslave	PVBusSlave
ARM CortexA55CT_CortexA76CT.DSU.shared_cache	PVCache
ARM CortexA55CT_CortexA76CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM CortexA55CT_CortexA76CT.DSU.shared_cache.upstream[1]	PVBusSlave

InstanceName	ComponentName
ARMCortexA55CT_CortexA76CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA55CT_CortexA76CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA55CT_CortexA76CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA55CT_CortexA76CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA55CT_CortexA76CT.MMAP	PVBusLogger
ARMCortexA55CT_CortexA76CT.MMAP.mapper	PVBusMapper
ARMCortexA55CT_CortexA76CT.ext_bus	PVBusLogger
ARMCortexA55CT_CortexA76CT.ext_bus.mapper	PVBusMapper
ARMCortexA55CT_CortexA76CT.gic_cpuif_decoder_cluster	GIcV3CPUInterfaceDecoder
ARMCortexA55CT_CortexA76CT.subcluster0	Subcluster_ARM_Cortex-A55
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0	ARM_Cortex-A55
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster1	Subcluster_ARM_Cortex-A76
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0	ARM_Cortex-A76
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster1.cpu1.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-244: ARMCortexA55CT_CortexA76CT MTI instances

InstanceName	ComponentName
ARMCortexA55CT_CortexA76CT	ARMv8Cluster
ARMCortexA55CT_CortexA76CT.AMU	PVBusLogger
ARMCortexA55CT_CortexA76CT.AMU.mapper	PVBusMapper
ARMCortexA55CT_CortexA76CT.DAP	PVBusLogger

InstanceName	ComponentName
ARMCortexA55CT_CortexA76CT.DAP.mapper	PVBusMapper
ARMCortexA55CT_CortexA76CT.DSU	DSU
ARMCortexA55CT_CortexA76CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA55CT_CortexA76CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA55CT_CortexA76CT.DSU.shared_cache	PVCache
ARMCortexA55CT_CortexA76CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA76CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA55CT_CortexA76CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA55CT_CortexA76CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA55CT_CortexA76CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA55CT_CortexA76CT.MMAP	PVBusLogger
ARMCortexA55CT_CortexA76CT.MMAP.mapper	PVBusMapper
ARMCortexA55CT_CortexA76CT.ext_bus	PVBusLogger
ARMCortexA55CT_CortexA76CT.ext_bus.mapper	PVBusMapper
ARMCortexA55CT_CortexA76CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0	ARM_Cortex-A55
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0	ARM_Cortex-A76
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l1licache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA76CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA55CT_CortexA76CT contains the following CADI targets:

- ARM_Cortex-A55
- ARM_Cortex-A76
- Cluster_ARM_Cortex-A55_Cortex-A76

- PVCache
- Subcluster_ARM_Cortex-A55
- Subcluster_ARM_Cortex-A76
- TlbCadi

About ARMCortexA55CT_CortexA76CT

The number of cores in each subcluster is configurable using the following parameters:

subcluster0.NUM_CORES

Possible values are 1-7 (ARMCortexA55CT).

subcluster1.NUM_CORES

Possible values are 1-4 (ARMCortexA76CT).

The total number of cores in the cluster cannot exceed 8.

Port arrays are expanded to the width needed for the maximum number of cores in each subcluster. Use the following port array indexes:

- `<port_name>[0-6]` for cores in `subcluster0`.
- `<port_name>[7-10]` for cores in `subcluster1`.



All instances in the Master cross trigger matrix port array, `cti[11]` must be connected, regardless of the `NUM_CORES` value used.

Core-specific parameters have the following prefixes:

- `subcluster0.cpu0` to `subcluster0.cpu6` identify cores in `subcluster0`.
- `subcluster1.cpu0` to `subcluster1.cpu3` identify cores in `subcluster1`.

For information about the cores in this model, see:

- [3.5.16 ARMCortexA55CT](#) on page 970.
- [3.5.28 ARMCortexA76CT](#) on page 1285.

Ports for ARMCortexA55CT_CortexA76CT

Table 3-245: Ports

Name	Protocol	Type	Description
aa64naa32[11]	Signal	Slave	Register width after reset.
acp_s	PVBus	Slave	AXI ACP slave port
AENDMP	Value_64	Slave	DynamiQ port to obtain end address of valid peripheral address range (exclusive).
ASTARTMP	Value_64	Slave	DynamiQ port to obtain start address of valid peripheral address range (inclusive).

Name	Protocol	Type	Description
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[11]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[11]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[11]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[11]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[11]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[11]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[11]	Signal	Master	Timer signals to SOC.
commirq[11]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[11]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cp15sdisable[11]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[11]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[11]	Signal	Slave	Disable cryptography extensions after reset.
cti[11]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[11]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[11]	Signal	Master	No power-down request.
dbgpwrupreq[11]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[11]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.

Name	Protocol	Type	Description
gicv3_redistributor_s[11]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[11]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[11]	PChannel	Slave	PChannels for cores
pmbirq[11]	Signal	Master	Interrupt signal from statistical profiling unit.
pmuirq[11]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[11]	Signal	Slave	Per core RAM Error Interrupt.
reset[11]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[11]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[11]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[11]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[11]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[11]	Signal	Slave	Virtualised FIQ.
vinithi[11]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[11]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L3 system and becomes coherent (assuming attributes are set correctly).
vsei[11]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA55CT_CortexA76CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster0.cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.VINITHI

Reset value of SCTL.R.V.

Type

bool

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x40000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.VINITHI

Reset value of SCTL.R.V.

Type

bool

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlbi invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.19 ARM CortexA55CT_CortexA78CT

ARM CortexA55CT_CortexA78CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-246: IP revisions support

Revision	Quality level
CortexA55 r1p0	Full support
CortexA78 r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `subcluster0.tlbi_stall_enabled`
- `subcluster1.tlbi_stall_enabled`

Iris and MTI instances for ARMCortexA55CT_CortexA78CT

This model has the following Iris instances:

Table 3-247: ARMCortexA55CT_CortexA78CT Iris instances

InstanceName	ComponentName
ARMCortexA55CT_CortexA78CT	Cluster_ARM_Cortex-A55_Cortex-A78
ARMCortexA55CT_CortexA78CT.AMU	PVBusLogger
ARMCortexA55CT_CortexA78CT.AMU.mapper	PVBusMapper
ARMCortexA55CT_CortexA78CT.DAP	PVBusLogger
ARMCortexA55CT_CortexA78CT.DAP.mapper	PVBusMapper
ARMCortexA55CT_CortexA78CT.DSU	DSU
ARMCortexA55CT_CortexA78CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA55CT_CortexA78CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache	PVCache
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA55CT_CortexA78CT.MMAP	PVBusLogger
ARMCortexA55CT_CortexA78CT.MMAP.mapper	PVBusMapper
ARMCortexA55CT_CortexA78CT.ext_bus	PVBusLogger
ARMCortexA55CT_CortexA78CT.ext_bus.mapper	PVBusMapper
ARMCortexA55CT_CortexA78CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA55CT_CortexA78CT.subcluster0	Subcluster_ARM_Cortex-A55
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0	ARM_Cortex-A55
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.dtlb	TlbCadi

InstanceName	ComponentName
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster1	Subcluster_ARM_Cortex-A78
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0	ARM_Cortex-A78
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster1.cpu1.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-248: ARMCortexA55CT_CortexA78CT MTI instances

InstanceName	ComponentName
ARMCortexA55CT_CortexA78CT	ARMv8Cluster
ARMCortexA55CT_CortexA78CT.AMU	PVBusLogger
ARMCortexA55CT_CortexA78CT.AMU.mapper	PVBusMapper
ARMCortexA55CT_CortexA78CT.DAP	PVBusLogger
ARMCortexA55CT_CortexA78CT.DAP.mapper	PVBusMapper
ARMCortexA55CT_CortexA78CT.DSU	DSU
ARMCortexA55CT_CortexA78CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA55CT_CortexA78CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache	PVCache
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA55CT_CortexA78CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA55CT_CortexA78CT.MMAP	PVBusLogger
ARMCortexA55CT_CortexA78CT.MMAP.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA55CT_CortexA78CT.ext_bus	PVBusLogger
ARMCortexA55CT_CortexA78CT.ext_bus.mapper	PVBusMapper
ARMCortexA55CT_CortexA78CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0	ARM_Cortex-A55
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0	ARM_Cortex-A78
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA55CT_CortexA78CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA55CT_CortexA78CT contains the following CADI targets:

- ARM_Cortex-A55
- ARM_Cortex-A78
- Cluster_ARM_Cortex-A55_Cortex-A78
- PVCache
- Subcluster_ARM_Cortex-A55
- Subcluster_ARM_Cortex-A78
- TlbCadi

About ARMCortexA55CT_CortexA78CT

The number of cores in each subcluster is configurable using the following parameters:

subcluster0.NUM_CORES

Possible values are 1-7 (ARMCortexA55CT).

subcluster1.NUM_CORES

Possible values are 1-4 (ARMCortexA78CT).

The total number of cores in the cluster cannot exceed 8.

Port arrays are expanded to the width needed for the maximum number of cores in each subcluster. Use the following port array indexes:

- `<port_name>[0-6]` for cores in `subcluster0`.
- `<port_name>[7-10]` for cores in `subcluster1`.



All instances in the Master cross trigger matrix port array, `cti[11]` must be connected, regardless of the `NUM_CORES` value used.

Core-specific parameters have the following prefixes:

- `subcluster0.cpu0` to `subcluster0.cpu6` identify cores in `subcluster0`.
- `subcluster1.cpu0` to `subcluster1.cpu3` identify cores in `subcluster1`.

For information about the cores in this model, see:

- [3.5.16 ARMCortexA55CT](#) on page 970.
- [3.5.32 ARMCortexA78CT](#) on page 1378.

Ports for ARMCortexA55CT_CortexA78CT

Table 3-249: Ports

Name	Protocol	Type	Description
<code>aa64naa32[11]</code>	Signal	Slave	Register width after reset.
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port
<code>AENDMP</code>	Value_64	Slave	DynamlQ port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	Value_64	Slave	DynamlQ port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[11]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>cfgte[11]</code>	Signal	Slave	This signal provides default exception handling state.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
<code>clusterid</code>	Value	Slave	The port reads the value in CPU ID register field, bits[11:14] of the MPIDR.
<code>clusterpmuirq</code>	Signal	Master	DynamlQ pmu irq

Name	Protocol	Type	Description
CNTHPIRQ[11]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[11]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[11]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[11]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[11]	Signal	Master	Timer signals to SOC.
commirq[11]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[11]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cp15sdisable[11]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[11]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[11]	Signal	Slave	Disable cryptography extensions after reset.
cti[11]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[11]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[11]	Signal	Master	No power-down request.
dbgpwrupreq[11]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[11]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[11]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[11]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[11]	PChannel	Slave	PChannels for cores
pmbirq[11]	Signal	Master	Interrupt signal from statistical profiling unit.
pmuirq[11]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.

Name	Protocol	Type	Description
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[11]	Signal	Slave	Per core RAM Error Interrupt.
reset[11]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[11]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[11]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[11]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[11]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[11]	Signal	Slave	Virtualised FIQ.
vinithi[11]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[11]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L3 system and becomes coherent (assuming attributes are set correctly).
vsei[11]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA55CT_CortexA78CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive (corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster0.cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x40000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.VINITHI

Reset value of SCTL.R.V.

Type

bool

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x0

subcluster1.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.20 ARMCortexA57x1CT

ARMCortexA57x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-250: IP revisions support

Revision	Quality level
r1p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA57x1CT

This model has the following Iris instances:

Table 3-251: ARMCortexA57x1CT Iris instances

InstanceName	ComponentName
ARMCortexA57x1CT	Cluster_ARM_Cortex-A57
ARMCortexA57x1CT.AMU	PVBusLogger
ARMCortexA57x1CT.AMU.mapper	PVBusMapper
ARMCortexA57x1CT.DAP	PVBusLogger
ARMCortexA57x1CT.DAP.mapper	PVBusMapper
ARMCortexA57x1CT.DSU	DSU
ARMCortexA57x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA57x1CT.MMAP	PVBusLogger
ARMCortexA57x1CT.MMAP.mapper	PVBusMapper
ARMCortexA57x1CT.acp_mapper	PVBusMapper
ARMCortexA57x1CT.cpu0	ARM_Cortex-A57
ARMCortexA57x1CT.cpu0.UTLB	TLB
ARMCortexA57x1CT.cpu0.dtlb	TlbCadi
ARMCortexA57x1CT.cpu0.l1dcache	PVCache
ARMCortexA57x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA57x1CT.cpu0.l1icache	PVCache
ARMCortexA57x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA57x1CT.ext_bus	PVBusLogger
ARMCortexA57x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA57x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA57x1CT.l2_cache	PVCache
ARMCortexA57x1CT.l2_cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA57x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA57x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-252: ARMCortexA57x1CT MTI instances

InstanceName	ComponentName
ARMCortexA57x1CT	ARMv8Cluster
ARMCortexA57x1CT.AMU	PVBusLogger
ARMCortexA57x1CT.AMU.mapper	PVBusMapper
ARMCortexA57x1CT.DAP	PVBusLogger
ARMCortexA57x1CT.DAP.mapper	PVBusMapper
ARMCortexA57x1CT.DSU	DSU
ARMCortexA57x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA57x1CT.MMAP	PVBusLogger
ARMCortexA57x1CT.MMAP.mapper	PVBusMapper
ARMCortexA57x1CT.acp_mapper	PVBusMapper
ARMCortexA57x1CT.cpu0	ARM_Cortex-A57
ARMCortexA57x1CT.cpu0.UTLB	TLB
ARMCortexA57x1CT.cpu0.l1dcache	PVCache
ARMCortexA57x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA57x1CT.cpu0.l1icache	PVCache
ARMCortexA57x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA57x1CT.ext_bus	PVBusLogger
ARMCortexA57x1CT.ext_bus.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA57x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA57x1CT.l2_cache	PVCache
ARMCortexA57x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA57x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA57x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA57x1CT contains the following CADI targets:

- ARM_Cortex-A57
- Cluster_ARM_Cortex-A57
- PVCache
- TlbCadi

About ARMCortexA57x1CT

- The following components also exist:
 - ARMCortexA57x2CT.
 - ARMCortexA57x3CT.
 - ARMCortexA57x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

- The cache latency parameters are only effective when you enable cache-state modeling.
- Timing annotation for transactions downstream of the cache and TLB models propagates through the models.

- This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).
- The system designer decides whether a debug APB ties the external debug view with other system views. In the model, use a `PVBusDecoder` to direct traffic to the correct port, `dev_debug_s` or `memorymapped_debug_s`.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- The value of the AArch64 PMCEID0_ELO register, and the AArch32 alias of this register, differs in the model from the TRM value. The model value reflects the model counters.
- The mechanisms for setting the affinity fields of the MPIDR. The RTL has two ports:
 - `CLUSTERIDAFF1[7:0]`.
 - `CLUSTERIDAFF2[7:0]`.

AFF1 sets the value of MPIDR bits[15:8] and AFF2 sets the value of MPIDR bits[23:16]. In contrast, the model has a single `CLUSTER_ID` port. This difference allows the setting of bits[23:8] of the MPIDR using bits[15:0] of the `CLUSTER_ID` value.

- The memory mapped debug registers have a view for cores and a view for external debug agents. In the model, these views require two PVBus ports. In hardware, the system designer decides how the implementation differentiates the views.
- In the model, a single `peer` event port combines the functionality of the `eventi` and `evento` signals in the RTL.
- The Generic Timers are Programmer's View (PV)-level abstractions: a model-specific protocol connects the `cntvalueb` port to the MemoryMappedCounterModule.
- The GIC CPU Interface is a PV-level abstraction: a model-specific protocol connects the GIC CPU Interface to the GIC Distributor.
- The Coresight™ *Cross Trigger Interface* (CTI) is a PV-level abstraction: the interface is a model-specific one.
- The model has no mechanism to read the internal memory that the Cache and TLB structures use, through the implementation defined region of the system coprocessor interface. This memory includes the RAM Index Register, IL1DATA Registers, DL1DATA Registers, and associated functionality.
- The model does not implement:
 - ETM registers.
 - The PMUEVENT bus.
 - The WARMRESETREQ signal. However, the warm reset code sequence (see the section *Code sequence to request a Warm reset as a result of RMR_ELx.RR* in the [Arm Architecture Reference Manual for A-profile architecture](#)) makes the model simulate a warm reset of the core.
 - The PMUSNAPSHOTREQ and PMUSNAPSHOTACK signals.

- The EXTERRIRQ and INTERRIRQ signals.
- Processor dynamic-retention signals.
- The SYSBARDISABLE signal.
- The DBGPWRDUP, DBGPWRUPREQ, DBGNOPWRDWN, and DBGRSTREQ debug power management signals.
- ECC and parity schemes are hardware-specific so are not supported.

Ports for ARMCortexA57x1CT

Table 3-253: Ports

Name	Protocol	Type	Description
aa64naa32[4]	Signal	Slave	Register width after reset.
acp_s	PVBus	Slave	AXI ACP slave port.
broadcastcachemaint	Signal	Slave	Enable broadcasting of cache maintenance operations to downstream caches.
broadcastinner	Signal	Slave	Enable broadcasting of Inner Shareable transactions.
broadcastouter	Signal	Slave	Enable broadcasting of Outer Shareable transactions.
cfgend[4]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[4]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clrexmonack	Signal	Master	Acknowledge handshake signal for the clrexmonreq signal
clrexmonreq	Signal	Slave	Signals the clearing of an external global exclusive monitor
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
CNTHPIRQ[4]	Signal	Master	The per-EL counter signal.
CNTPNSIRQ[4]	Signal	Master	The per-EL counter signal.
CNTPSIRQ[4]	Signal	Master	The per-EL counter signal.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	The per-EL counter signal.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
commrx[4]	Signal	Master	Receive portion of Data Transfer Register full.
commtx[4]	Signal	Master	Transmit portion of Data Transfer Register empty.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers.

Name	Protocol	Type	Description
cpuporeset[4]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgack[4]	Signal	Master	External debug interface.
dbgen[4]	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	These signals relate to core power down.
dbgpwrupreq[4]	Signal	Master	These signals relate to core power down.
dev_debug_s	PVBus	Slave	External debug interface.
edbgrq[4]	Signal	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
l2flushdone	Signal	Master	Flush of L2 memory system complete
l2flushreq	Signal	Slave	Request flush of L2 memory system.
l2reset	Signal	Slave	Reset the shared L2 memory system controller.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[4]	Signal	Slave	External debug interface.
periphbase	Value_64	Slave	This port sets the base address of private peripheral region.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
rei[4]	Signal	Slave	Individual processor RAM Error Interrupt signal input.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sei[4]	Signal	Slave	Per core System Error physical pins.

Name	Protocol	Type	Description
smpen[4]	Signal	Master	This signals AMP or SMP mode for each core.
spiden[4]	Signal	Slave	External debug interface.
spniden[4]	Signal	Slave	External debug interface.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state.
standbywfi12	Signal	Master	Indicate that all the individual processors and the L2 systems are in a WFI state.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtual FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtual IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Processor Virtual System Error Interrupt request.

Parameters for ARM Cortex A57x1CT

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x22000000

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

ptw_latency	Page table walker latency for TA (Timing Annotation), expressed in simulation ticks
Type	int
Default value	0x0
tlb_latency	TLB latency for TA (Timing Annotation), expressed in simulation ticks
Type	int
Default value	0x0
tlbi_stall_enabled	If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.
Type	bool
Default value	0x0
walk_cache_latency	Walk cache latency for TA (Timing Annotation), expressed in simulation ticks
Type	int
Default value	0x0

3.5.21 ARMCortexA65AECT

ARMCortexA65AECT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-254: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `tlbi_stall_enabled`

Iris and MTI instances for ARM Cortex-A65AE

This model has the following Iris instances:

Table 3-255: ARM Cortex-A65AE Iris instances

InstanceName	ComponentName
ARM Cortex-A65AE	Cluster_ARM_Cortex-A65AE
ARM Cortex-A65AE.AMU	PVBusLogger
ARM Cortex-A65AE.AMU.mapper	PVBusMapper
ARM Cortex-A65AE.DAP	PVBusLogger
ARM Cortex-A65AE.DAP.mapper	PVBusMapper
ARM Cortex-A65AE.DSU	DSU
ARM Cortex-A65AE.DSU.l3_flusher	AsyncCacheFlushUnit
ARM Cortex-A65AE.DSU.mpam_busslave	PVBusSlave
ARM Cortex-A65AE.DSU.shared_cache	PVCache
ARM Cortex-A65AE.DSU.shared_cache.upstream[0]	PVBusSlave
ARM Cortex-A65AE.DSU.shared_cache.upstream[1]	PVBusSlave
ARM Cortex-A65AE.DSU.shared_cache.upstream[2]	PVBusSlave
ARM Cortex-A65AE.DSU.shared_cache.upstream[3]	PVBusSlave
ARM Cortex-A65AE.DSU.shared_cache.upstream[4]	PVBusSlave
ARM Cortex-A65AE.DSU.shared_cache.upstream[5]	PVBusSlave
ARM Cortex-A65AE.MMAP	PVBusLogger
ARM Cortex-A65AE.MMAP.mapper	PVBusMapper
ARM Cortex-A65AE.cpu0.dtlb	TlbCadi
ARM Cortex-A65AE.cpu0.thread0	ARM_Cortex-A65AE
ARM Cortex-A65AE.cpu0.thread0.UTLB	TLB
ARM Cortex-A65AE.cpu0.thread0.l1dcache	PVCache
ARM Cortex-A65AE.cpu0.thread0.l1dcache.upstream[0]	PVBusSlave
ARM Cortex-A65AE.cpu0.thread0.l1icache	PVCache
ARM Cortex-A65AE.cpu0.thread0.l1icache.upstream[0]	PVBusSlave
ARM Cortex-A65AE.cpu0.thread0.l2cache	PVCache
ARM Cortex-A65AE.cpu0.thread0.l2cache.upstream[0]	PVBusSlave
ARM Cortex-A65AE.cpu0.thread0.l2cache.upstream[1]	PVBusSlave
ARM Cortex-A65AE.cpu0.thread1	ARM_Cortex-A65AE
ARM Cortex-A65AE.cpu0.thread1.UTLB	TLB
ARM Cortex-A65AE.cpu1.dtlb	TlbCadi
ARM Cortex-A65AE.cpu1.thread0	ARM_Cortex-A65AE

InstanceName	ComponentName
ARMCortexA65AECT.cpu1.thread0.UTLB	TLB
ARMCortexA65AECT.cpu1.thread0.l1dcache	PVCache
ARMCortexA65AECT.cpu1.thread0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT.cpu1.thread0.l1icache	PVCache
ARMCortexA65AECT.cpu1.thread0.l1icache.upstream[0]	PVBusSlave
ARMCortexA65AECT.cpu1.thread0.l2cache	PVCache
ARMCortexA65AECT.cpu1.thread0.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT.cpu1.thread0.l2cache.upstream[1]	PVBusSlave
ARMCortexA65AECT.cpu1.thread1	ARM_Cortex-A65AE
ARMCortexA65AECT.cpu1.thread1.UTLB	TLB
ARMCortexA65AECT.ext_bus	PVBusLogger
ARMCortexA65AECT.ext_bus.mapper	PVBusMapper
ARMCortexA65AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-256: ARMCortexA65AECT MTI instances

InstanceName	ComponentName
ARMCortexA65AECT	ARMv8Cluster
ARMCortexA65AECT.AMU	PVBusLogger
ARMCortexA65AECT.AMU.mapper	PVBusMapper
ARMCortexA65AECT.DAP	PVBusLogger
ARMCortexA65AECT.DAP.mapper	PVBusMapper
ARMCortexA65AECT.DSU	DSU
ARMCortexA65AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA65AECT.DSU.mpam_busslave	PVBusSlave
ARMCortexA65AECT.DSU.shared_cache	PVCache
ARMCortexA65AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA65AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA65AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA65AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA65AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA65AECT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA65AECT.MMAP	PVBusLogger
ARMCortexA65AECT.MMAP.mapper	PVBusMapper
ARMCortexA65AECT.cpu0.thread0	ARM_Cortex-A65AE
ARMCortexA65AECT.cpu0.thread0.UTLB	TLB
ARMCortexA65AECT.cpu0.thread0.l1dcache	PVCache
ARMCortexA65AECT.cpu0.thread0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT.cpu0.thread0.l1icache	PVCache

InstanceName	ComponentName
ARMCortexA65AECT.cpu0.thread0.l1licache.upstream[0]	PVBusSlave
ARMCortexA65AECT.cpu0.thread0.l2cache	PVCache
ARMCortexA65AECT.cpu0.thread0.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT.cpu0.thread0.l2cache.upstream[1]	PVBusSlave
ARMCortexA65AECT.cpu0.thread1	ARM_Cortex-A65AE
ARMCortexA65AECT.cpu0.thread1.UTLB	TLB
ARMCortexA65AECT.cpu1.thread0	ARM_Cortex-A65AE
ARMCortexA65AECT.cpu1.thread0.UTLB	TLB
ARMCortexA65AECT.cpu1.thread0.l1dcache	PVCache
ARMCortexA65AECT.cpu1.thread0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT.cpu1.thread0.l1licache	PVCache
ARMCortexA65AECT.cpu1.thread0.l1licache.upstream[0]	PVBusSlave
ARMCortexA65AECT.cpu1.thread0.l2cache	PVCache
ARMCortexA65AECT.cpu1.thread0.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT.cpu1.thread0.l2cache.upstream[1]	PVBusSlave
ARMCortexA65AECT.cpu1.thread1	ARM_Cortex-A65AE
ARMCortexA65AECT.cpu1.thread1.UTLB	TLB
ARMCortexA65AECT.ext_bus	PVBusLogger
ARMCortexA65AECT.ext_bus.mapper	PVBusMapper
ARMCortexA65AECT.gic_cpuif_decoder_cluster	GIcV3CPUInterfaceDecoder

ARMCortexA65AECT contains the following CADI targets:

- ARM_Cortex-A65AE
- Cluster_ARM_Cortex-A65AE
- PVCache
- TlbCadi

About ARMCortexA65AECT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- `BROADCASTCACHEMAINTPOU` pin.
- `COREINSTRET`, `COREINSTRRUN`, or `nPMBIRQ` signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals `DBGPWRDUP` and `DBGIRSTREQ` are not implemented, but `DBGPWRUPREQ` and `DBGNOPWRDWN` are implemented.
- Cache stashing capability.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.
- Split/Lock is supported but with the limitations described in the AE-specific features implemented section.

This model has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following models also exist, with a fixed number of cores per cluster:

- `ARMCortexA65AECTx2CT`.
- `ARMCortexA65AECTx4CT`.
- `ARMCortexA65AECTx6CT`.
- `ARMCortexA65AECTx8CT`.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-7).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

AE-specific features implemented

The only AE-specific features implemented in the model are Split-Lock and Lock-Step, with the following limitations:

- The implementation of Split-Lock and Lock-Step in the model is simply a halving of the number of cores available.
- Lock-step is implemented by a parameter that can be true or false for the entire execution only, rather than by a signal that can be enabled during execution.

As the model does not implement true Split-Lock functionality, the ports and parameters are contiguous. For example, for a 4-core CPU with Split-Lock enabled, `cpu0` and `cpu1` identify the available cores and associated ports, not `cpu0` and `cpu2`.

- Hybrid mode is not modeled in the DSU.

Ports for ARMCortexA65AECT

Table 3-257: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port
<code>AENDMP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[16]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgte[16]</code>	Signal	Slave	This signal provides default exception handling state.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>clusterid</code>	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
<code>clusterpmuirq</code>	Signal	Master	DynamlQ pmu irq
<code>CNTHPIRQ[16]</code>	Signal	Master	Timer signals to SOC.
<code>CNTHVIRQ[16]</code>	Signal	Master	Timer signals to SOC.
<code>CNTPNSIRQ[16]</code>	Signal	Master	Timer signals to SOC.
<code>CNTPSIRQ[16]</code>	Signal	Master	Timer signals to SOC.
<code>cntvalueb</code>	CounterInterface	Slave	Interface to SoC level counter module.
<code>CNTVIRQ[16]</code>	Signal	Master	Timer signals to SOC.
<code>commirq[16]</code>	Signal	Master	Interrupt signal from debug communications channel.
<code>core_clk_in[8]</code>	ClockSignal	Slave	The clock signal connected to the <code>core_clk_in</code> port is used to determine the rate at which each core executes instructions.
<code>cpuporeset[8]</code>	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
<code>cryptodisable[16]</code>	Signal	Slave	Disable cryptography extensions after reset.
<code>cti[16]</code>	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
<code>ctidbgirq[16]</code>	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.

Name	Protocol	Type	Description
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[16]	Signal	Master	These signals relate to core power down.
dbgpwrupreq[16]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[16]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[16]	GICv3Comms	Slave	GICv3 AXI-stream port per thread.
irq[16]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[8]	PChannel	Slave	PChannels for cores
pmuirq[16]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[16]	Signal	Slave	Per core RAM Error Interrupt.
reset[8]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[16]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[16]	Signal	Slave	Per core virtual System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[16]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[16]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[16]	Signal	Slave	Virtualised FIQ.

Name	Protocol	Type	Description
vinithi[16]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[16]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[16]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA65AECT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x2

cluster_patch_level

Cosmetic change to patch number in CLUSTERIDR. Corresponds to the Y in rXpY.

Type

int

Default value

0x0

cluster_revision_number

Cosmetic change to revision number in CLUSTERIDR, Corresponds to the X in rXpY.

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.enable_single_thread_at_reset

Enable single thread after reset and keep other thread in reset

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x40000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.threadY.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.threadY.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.threadY.MPIDR-override

Override MPIDR value. A nonzero value will override the MT,cluster,cpu,thread ID bits in MPIDR.

Type

int

Default value

0x0

cpuX.threadY.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.threadY.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_lock_step

Whether the core is configured in Dual Core Lock Step mode (FEAT_DCLS)

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit

or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-read_latency`

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l3cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-size`

L3 Cache size in bytes.

Type

int

Default value

0x400000

`l3cache-snoop_data_transfer_latency`

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-snoop_issue_latency`

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.22 ARM CortexA65AECT_CortexA76AECT

ARM CortexA65AECT_CortexA76AECT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-258: IP revisions support

Revision	Quality level
CortexA65AE r0p0	Full support
CortexA76AE r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- subcluster0.tlbi_stall_enabled
- subcluster1.tlbi_stall_enabled

Iris and MTI instances for ARM CortexA65AECT_CortexA76AECT

This model has the following Iris instances:

Table 3-259: ARM CortexA65AECT_CortexA76AECT Iris instances

InstanceName	ComponentName
ARM CortexA65AECT_CortexA76AECT	Cluster_ARM_Cortex-A65AE_Cortex-A76AE
ARM CortexA65AECT_CortexA76AECT.AMU	PVBusLogger
ARM CortexA65AECT_CortexA76AECT.AMU.mapper	PVBusMapper
ARM CortexA65AECT_CortexA76AECT.DAP	PVBusLogger
ARM CortexA65AECT_CortexA76AECT.DAP.mapper	PVBusMapper
ARM CortexA65AECT_CortexA76AECT.DSU	DSU
ARM CortexA65AECT_CortexA76AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM CortexA65AECT_CortexA76AECT.DSU.mpam_busslave	PVBusSlave
ARM CortexA65AECT_CortexA76AECT.DSU.shared_cache	PVCache
ARM CortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM CortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARM CortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARM CortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[3]	PVBusSlave

InstanceName	ComponentName
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[7]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[8]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[9]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.MMAP	PVBusLogger
ARMCortexA65AECT_CortexA76AECT.MMAP.mapper	PVBusMapper
ARMCortexA65AECT_CortexA76AECT.ext_bus	PVBusLogger
ARMCortexA65AECT_CortexA76AECT.ext_bus.mapper	PVBusMapper
ARMCortexA65AECT_CortexA76AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA65AECT_CortexA76AECT.subcluster0	Subcluster_ARM_Cortex-A65AE
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0	ARM_Cortex-A65AE
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l1dcache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l1icache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l1icache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l2cache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l2cache.upstream[1]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread1	ARM_Cortex-A65AE
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread1.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.dtlb	TlbCadi
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0	ARM_Cortex-A65AE
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l1dcache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l1icache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l1icache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l2cache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l2cache.upstream[1]	PVBusSlave

InstanceName	ComponentName
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread1	ARM_Cortex-A65AE
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread1.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster1	Subcluster_ARM_Cortex-A76AE
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0	ARM_Cortex-A76AE
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1	ARM_Cortex-A76AE
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l1dcache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l1icache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l1icache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l2cache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l2cache.upstream[1]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu2.dtlb	TlbCadi
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu3.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-260: ARMCortexA65AECT_CortexA76AECT MTI instances

InstanceName	ComponentName
ARMCortexA65AECT_CortexA76AECT	ARMv8Cluster
ARMCortexA65AECT_CortexA76AECT.AMU	PVBusLogger
ARMCortexA65AECT_CortexA76AECT.AMU.mapper	PVBusMapper
ARMCortexA65AECT_CortexA76AECT.DAP	PVBusLogger
ARMCortexA65AECT_CortexA76AECT.DAP.mapper	PVBusMapper
ARMCortexA65AECT_CortexA76AECT.DSU	DSU
ARMCortexA65AECT_CortexA76AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA65AECT_CortexA76AECT.DSU.mpam_busslave	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache	PVCache
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[2]	PVBusSlave

InstanceName	ComponentName
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[7]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[8]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.DSU.shared_cache.upstream[9]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.MMAP	PVBusLogger
ARMCortexA65AECT_CortexA76AECT.MMAP.mapper	PVBusMapper
ARMCortexA65AECT_CortexA76AECT.ext_bus	PVBusLogger
ARMCortexA65AECT_CortexA76AECT.ext_bus.mapper	PVBusMapper
ARMCortexA65AECT_CortexA76AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0	ARM_Cortex-A65AE
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l1dcache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l1licache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l1licache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l2cache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread0.l2cache.upstream[1]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread1	ARM_Cortex-A65AE
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu0.thread1.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0	ARM_Cortex-A65AE
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l1dcache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l1licache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l1licache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l2cache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread0.l2cache.upstream[1]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread1	ARM_Cortex-A65AE
ARMCortexA65AECT_CortexA76AECT.subcluster0.cpu1.thread1.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0	ARM_Cortex-A76AE
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l1licache	PVCache

InstanceName	ComponentName
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1	ARM_Cortex-A76AE
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.UTLB	TLB
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l1dcache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l1licache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l1licache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l2cache	PVCache
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l2cache.upstream[0]	PVBusSlave
ARMCortexA65AECT_CortexA76AECT.subcluster1.cpu1.l2cache.upstream[1]	PVBusSlave

ARMCortexA65AECT_CortexA76AECT contains the following CADI targets:

- ARM_Cortex-A65AE
- ARM_Cortex-A76AE
- Cluster_ARM_Cortex-A65AE_Cortex-A76AE
- PVCache
- Subcluster_ARM_Cortex-A65AE
- Subcluster_ARM_Cortex-A76AE
- TlbCadi

About ARMCortexA65AECT_CortexA76AECT

The number of cores in each subcluster is configurable using the following parameters:

subcluster0.NUM_CORES

Possible values are 2-6 (ARMCortexA65AECT).

subcluster1.NUM_CORES

Possible values are 2-4 (ARMCortexA76AECT).

The total number of cores in the cluster cannot exceed 8.

Port arrays are expanded to the width needed for the maximum number of cores in each subcluster. Use the following port array indexes:

- `<port_name>[0-11]` for threads in `subcluster0`. For example:
 - `<port_name>[0]` is a port for `subcluster0.cpu0.thread0`.
 - `<port_name>[1]` is a port for `subcluster0.cpu0.thread1`.
 - `<port_name>[2]` is a port for `subcluster0.cpu1.thread0`.

- `<port_name>[12-15]` for cores in `subcluster1`.



All instances in the Master cross trigger matrix port array `cti[16]` must be connected, regardless of the `NUM_CORES` value used.

Core-specific parameters have the following prefixes:

- `subcluster0.cpu0` to `subcluster0.cpu5` identify cores in `subcluster0`.
- `subcluster1.cpu0` to `subcluster1.cpu3` identify cores in `subcluster1`.

For information about the cores in this model, see:

- [3.5.21 ARMCortexA65AECT](#) on page 1116.
- [3.5.27 ARMCortexA76AECT](#) on page 1261.

AE-specific features implemented

The only AE-specific features implemented in the model are Split-Lock and Lock-Step, with the following limitations:

- The implementation of Split-Lock and Lock-Step in the model is simply a halving of the number of cores available.
- Lock-step is implemented by a parameter that can be true or false for the entire execution only, rather than by a signal that can be enabled during execution.

As the model does not implement true Split-Lock functionality, the ports and parameters are contiguous. For example, for a 4-core CPU with Split-Lock enabled, `cpu0` and `cpu1` identify the available cores and associated ports, not `cpu0` and `cpu2`.

- Hybrid mode is not modeled in the DSU.

Ports for ARMCortexA65AECT_CortexA76AECT

Table 3-261: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port
<code>AENDMP</code>	Value_64	Slave	DynamlQ port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	Value_64	Slave	DynamlQ port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[16]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgte[16]</code>	Signal	Slave	This signal provides default exception handling state.

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[16]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[16]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[16]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[16]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[16]	Signal	Master	Timer signals to SOC.
commirq[16]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[10]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cpuporeset[10]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[16]	Signal	Slave	Disable cryptography extensions after reset.
cti[16]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[16]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[16]	Signal	Master	No power-down request.
dbgprupreq[16]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[16]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[16]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[16]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[10]	PChannel	Slave	PChannels for cores
pmbirq[16]	Signal	Master	Interrupt signal from statistical profiling unit.
pmuirq[16]	Signal	Master	Interrupt signal from performance monitoring unit.

Name	Protocol	Type	Description
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[16]	Signal	Slave	Per core RAM Error Interrupt.
reset[10]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[16]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[16]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[16]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[16]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[16]	Signal	Slave	Virtualised FIQ.
vinithi[16]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[16]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L3 system and becomes coherent (assuming attributes are set correctly).
vsei[16]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM CortexA65AECT_CortexA76AECT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_lock_step

Whether the core is configured in Dual Core Lock Step mode (FEAT_DCLS)

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x100000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x2

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.enable_single_thread_at_reset

Enable single thread after reset and keep other thread in reset

Type

bool

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x40000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value

subcluster0.cpuX.semihosting-enable

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.threadY.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.threadY.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster0.cpuX.threadY.MPIDR-override

Override MPIDR value. A nonzero value will override the MT,cluster,cpu,thread ID bits in MPIDR.

Type

int

Default value

0x0

subcluster0.cpuX.threadY.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.threadY.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be

used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

subcluster0.reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlbi invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x2

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.VINITHI

Reset value of SCTL.R.V.

Type

bool

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size `dcache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. `dcache-write_access_latency` must be set to 0 for per-byte latencies to be applied. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster1.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

subcluster1.reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlbi invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.23 ARMCortexA65CT

ARMCortexA65CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-262: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `tlbi_stall_enabled`

Iris and MTI instances for ARM Cortex-A65CT

This model has the following Iris instances:

Table 3-263: ARM Cortex-A65CT Iris instances

InstanceName	ComponentName
ARM Cortex-A65CT	Cluster_ARM_Cortex-A65
ARM Cortex-A65CT.AMU	PVBusLogger
ARM Cortex-A65CT.AMU.mapper	PVBusMapper
ARM Cortex-A65CT.DAP	PVBusLogger
ARM Cortex-A65CT.DAP.mapper	PVBusMapper
ARM Cortex-A65CT.DSU	DSU
ARM Cortex-A65CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM Cortex-A65CT.DSU.mpam_busslave	PVBusSlave
ARM Cortex-A65CT.DSU.shared_cache	PVCache
ARM Cortex-A65CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM Cortex-A65CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARM Cortex-A65CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARM Cortex-A65CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARM Cortex-A65CT.MMAP	PVBusLogger
ARM Cortex-A65CT.MMAP.mapper	PVBusMapper
ARM Cortex-A65CT.cpu0.dtlb	TlbCadi
ARM Cortex-A65CT.cpu0.thread0	ARM_Cortex-A65
ARM Cortex-A65CT.cpu0.thread0.UTLB	TLB
ARM Cortex-A65CT.cpu0.thread0.l1dcache	PVCache
ARM Cortex-A65CT.cpu0.thread0.l1dcache.upstream[0]	PVBusSlave
ARM Cortex-A65CT.cpu0.thread0.l1licache	PVCache
ARM Cortex-A65CT.cpu0.thread0.l1licache.upstream[0]	PVBusSlave
ARM Cortex-A65CT.cpu0.thread0.l2cache	PVCache
ARM Cortex-A65CT.cpu0.thread0.l2cache.upstream[0]	PVBusSlave
ARM Cortex-A65CT.cpu0.thread0.l2cache.upstream[1]	PVBusSlave
ARM Cortex-A65CT.cpu0.thread1	ARM_Cortex-A65
ARM Cortex-A65CT.cpu0.thread1.UTLB	TLB
ARM Cortex-A65CT.ext_bus	PVBusLogger
ARM Cortex-A65CT.ext_bus.mapper	PVBusMapper
ARM Cortex-A65CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-264: ARMCortexA65CT MTI instances

InstanceName	ComponentName
ARMCortexA65CT	ARMv8Cluster
ARMCortexA65CT.AMU	PVBusLogger
ARMCortexA65CT.AMU.mapper	PVBusMapper
ARMCortexA65CT.DAP	PVBusLogger
ARMCortexA65CT.DAP.mapper	PVBusMapper
ARMCortexA65CT.DSU	DSU
ARMCortexA65CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA65CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA65CT.DSU.shared_cache	PVCache
ARMCortexA65CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA65CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA65CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA65CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA65CT.MMAP	PVBusLogger
ARMCortexA65CT.MMAP.mapper	PVBusMapper
ARMCortexA65CT.cpu0.thread0	ARM_Cortex-A65
ARMCortexA65CT.cpu0.thread0.UTLB	TLB
ARMCortexA65CT.cpu0.thread0.l1dcache	PVCache
ARMCortexA65CT.cpu0.thread0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA65CT.cpu0.thread0.l1licache	PVCache
ARMCortexA65CT.cpu0.thread0.l1licache.upstream[0]	PVBusSlave
ARMCortexA65CT.cpu0.thread0.l2cache	PVCache
ARMCortexA65CT.cpu0.thread0.l2cache.upstream[0]	PVBusSlave
ARMCortexA65CT.cpu0.thread0.l2cache.upstream[1]	PVBusSlave
ARMCortexA65CT.cpu0.thread1	ARM_Cortex-A65
ARMCortexA65CT.cpu0.thread1.UTLB	TLB
ARMCortexA65CT.ext_bus	PVBusLogger
ARMCortexA65CT.ext_bus.mapper	PVBusMapper
ARMCortexA65CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA65CT contains the following CADI targets:

- ARM_Cortex-A65
- Cluster_ARM_Cortex-A65
- PVCache
- TlbCadi

About ARMCortexA65CT

The model supports the following features:

- DynamiQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- DynamiQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals DBGWRUP and DBGRSTREQ are not implemented, but DBGWRUPREQ and DBGNOFWRDWN are implemented.
- Cache stashing capability.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.

This component has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following components also exist, with a fixed number of cores per cluster:

- ARMCortexA65x1CT.
- ARMCortexA65x2CT.
- ARMCortexA65x3CT.
- ARMCortexA65x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA65CT

Table 3-265: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
AENDMP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTARTMP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[16]	Signal	Slave	This signal is for EE bit initialisation.
cfgte[16]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[16]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[16]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[16]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[16]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[16]	Signal	Master	Timer signals to SOC.
commirq[16]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[8]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cpuporeset[8]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[16]	Signal	Slave	Disable cryptography extensions after reset.
cti[16]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[16]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[16]	Signal	Master	These signals relate to core power down.
dbgpwrupreq[16]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.

Name	Protocol	Type	Description
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[16]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[16]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[16]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[8]	PChannel	Slave	PChannels for cores
pmuirq[16]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[16]	Signal	Slave	Per core RAM Error Interrupt.
reset[8]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[16]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[16]	Signal	Slave	Per core virtual System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynaMLQ.
ticks[16]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[16]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[16]	Signal	Slave	Virtualised FIQ.
vinithi[16]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[16]	Signal	Slave	Virtualised IRQ.

Name	Protocol	Type	Description
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[16]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A65CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAIN

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cluster_patch_level

Cosmetic change to patch number in CLUSTERIDR. Corresponds to the Y in rXpY.

Type

int

Default value

0x0

cluster_revision_number

Cosmetic change to revision number in CLUSTERIDR, Corresponds to the X in rXpY.

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.enable_single_thread_at_reset

Enable single thread after reset and keep other thread in reset

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x40000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.threadY.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.threadY.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.threadY.MPIDR-override

Override MPIDR value. A nonzero value will override the MT,cluster,cpu,thread ID bits in MPIDR.

Type

int

Default value

0x0

cpuX.threadY.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.threadY.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x400000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.24 ARMCortexA72x1CT

ARMCortexA72x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-266: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA72x1CT

This model has the following Iris instances:

Table 3-267: ARMCortexA72x1CT Iris instances

InstanceName	ComponentName
ARMCortexA72x1CT	Cluster_ARM_Cortex-A72
ARMCortexA72x1CT.AMU	PVBusLogger
ARMCortexA72x1CT.AMU.mapper	PVBusMapper
ARMCortexA72x1CT.DAP	PVBusLogger
ARMCortexA72x1CT.DAP.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA72x1CT.DSU	DSU
ARMCortexA72x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA72x1CT.MMAP	PVBusLogger
ARMCortexA72x1CT.MMAP.mapper	PVBusMapper
ARMCortexA72x1CT.acp_mapper	PVBusMapper
ARMCortexA72x1CT.cpu0	ARM_Cortex-A72
ARMCortexA72x1CT.cpu0.UTLB	TLB
ARMCortexA72x1CT.cpu0.dtlb	TlbCadi
ARMCortexA72x1CT.cpu0.l1dcache	PVCache
ARMCortexA72x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA72x1CT.cpu0.l1licache	PVCache
ARMCortexA72x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA72x1CT.ext_bus	PVBusLogger
ARMCortexA72x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA72x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA72x1CT.l2_cache	PVCache
ARMCortexA72x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA72x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-268: ARMCortexA72x1CT MTI instances

InstanceName	ComponentName
ARMCortexA72x1CT	ARMv8Cluster

InstanceName	ComponentName
ARMCortexA72x1CT.AMU	PVBusLogger
ARMCortexA72x1CT.AMU.mapper	PVBusMapper
ARMCortexA72x1CT.DAP	PVBusLogger
ARMCortexA72x1CT.DAP.mapper	PVBusMapper
ARMCortexA72x1CT.DSU	DSU
ARMCortexA72x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA72x1CT.MMAP	PVBusLogger
ARMCortexA72x1CT.MMAP.mapper	PVBusMapper
ARMCortexA72x1CT.acp_mapper	PVBusMapper
ARMCortexA72x1CT.cpu0	ARM_Cortex-A72
ARMCortexA72x1CT.cpu0.UTLB	TLB
ARMCortexA72x1CT.cpu0.l1dcache	PVCache
ARMCortexA72x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA72x1CT.cpu0.l1licache	PVCache
ARMCortexA72x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA72x1CT.ext_bus	PVBusLogger
ARMCortexA72x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA72x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA72x1CT.l2_cache	PVCache
ARMCortexA72x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA72x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA72x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA72x1CT contains the following CADI targets:

- ARM_Cortex-A72
- Cluster_ARM_Cortex-A72
- PVCache
- TlbCadi

About ARM Cortex A72x1CT

- The following components also exist:
 - ARM Cortex A72x2CT.
 - ARM Cortex A72x3CT.
 - ARM Cortex A72x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

- The cache latency cluster parameters are only effective when you enable cache-state modeling.
- Timing annotation for transactions downstream of the cache and TLB models propagates through the cache and TLB models.
- ECC and parity schemes are hardware-specific so are not supported.
- This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARM Cortex A72x1CT

Table 3-269: Ports

Name	Protocol	Type	Description
aa64naa32[4]	Signal	Slave	Register width after reset.
acp_s	PVBus	Slave	AXI ACP slave port.
broadcastcachemaint	Signal	Slave	Enable broadcasting of cache maintenance operations to downstream caches.
broadcastinner	Signal	Slave	Enable broadcasting of Inner Shareable transactions.
broadcastouter	Signal	Slave	Enable broadcasting of Outer Shareable transactions.
cfgend[4]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[4]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
clrexmonack	Signal	Master	Acknowledge handshake signal for the <code>clrexmonreq</code> signal
clrexmonreq	Signal	Slave	Signals the clearing of an external global exclusive monitor

Name	Protocol	Type	Description
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
CNTHPIRQ[4]	Signal	Master	The per-EL counter signal.
CNTPNSIRQ[4]	Signal	Master	The per-EL counter signal.
CNTPSIRQ[4]	Signal	Master	The per-EL counter signal.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module
CNTVIRQ[4]	Signal	Master	The per-EL counter signal.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
commrx[4]	Signal	Master	Receive portion of Data Transfer Register full.
commtx[4]	Signal	Master	Transmit portion of Data Transfer Register empty.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[4]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgack[4]	Signal	Master	External debug interface.
dbgen[4]	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	These signals relate to core power down.
dbgpwrupreq[4]	Signal	Master	These signals relate to core power down.
dev_debug_s	PVBus	Slave	External debug interface.
edbgrq[4]	Signal	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
l2flushdone	Signal	Master	Flush of L2 memory system complete.
l2flushreq	Signal	Slave	Request flush of L2 memory system.
l2reset	Signal	Slave	Reset the shared L2 memory system controller.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[4]	Signal	Slave	External debug interface.
periphbase	Value_64	Slave	This port sets the base address of private peripheral region.

Name	Protocol	Type	Description
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
rei[4]	Signal	Slave	Individual processor RAM Error Interrupt signal input.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sei[4]	Signal	Slave	Per core System Error physical pins.
smpen[4]	Signal	Master	This signals AMP or SMP mode for each core.
spiden[4]	Signal	Slave	External debug interface.
spniden[4]	Signal	Slave	External debug interface.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state.
standbywfil2	Signal	Master	Indicate that all the individual processors and the L2 systems are in a WFI state.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtual FIQ
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtual IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Processor Virtual System Error Interrupt request.

Parameters for ARM Cortex A72x1CT

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x22000000

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.25 ARMCortexA73x1CT

ARMCortexA73x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-270: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA73x1CT

This model has the following Iris instances:

Table 3-271: ARMCortexA73x1CT Iris instances

InstanceName	ComponentName
ARMCortexA73x1CT	Cluster_ARM_Cortex-A73
ARMCortexA73x1CT.AMU	PVBusLogger
ARMCortexA73x1CT.AMU.mapper	PVBusMapper
ARMCortexA73x1CT.DAP	PVBusLogger
ARMCortexA73x1CT.DAP.mapper	PVBusMapper
ARMCortexA73x1CT.DSU	DSU
ARMCortexA73x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA73x1CT.MMAP	PVBusLogger
ARMCortexA73x1CT.MMAP.mapper	PVBusMapper
ARMCortexA73x1CT.acp_mapper	PVBusMapper
ARMCortexA73x1CT.cpu0	ARM_Cortex-A73
ARMCortexA73x1CT.cpu0.UTLB	TLB
ARMCortexA73x1CT.cpu0.dtlb	TlbCadi
ARMCortexA73x1CT.cpu0.l1dcache	PVCache
ARMCortexA73x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA73x1CT.cpu0.l1licache	PVCache
ARMCortexA73x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA73x1CT.ext_bus	PVBusLogger
ARMCortexA73x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA73x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA73x1CT.l2_cache	PVCache
ARMCortexA73x1CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[12]	PVBusSlave

InstanceName	ComponentName
ARMCortexA73x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA73x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-272: ARMCortexA73x1CT MTI instances

InstanceName	ComponentName
ARMCortexA73x1CT	ARMv8Cluster
ARMCortexA73x1CT.AMU	PVBusLogger
ARMCortexA73x1CT.AMU.mapper	PVBusMapper
ARMCortexA73x1CT.DAP	PVBusLogger
ARMCortexA73x1CT.DAP.mapper	PVBusMapper
ARMCortexA73x1CT.DSU	DSU
ARMCortexA73x1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA73x1CT.MMAP	PVBusLogger
ARMCortexA73x1CT.MMAP.mapper	PVBusMapper
ARMCortexA73x1CT.acp_mapper	PVBusMapper
ARMCortexA73x1CT.cpu0	ARM_Cortex-A73
ARMCortexA73x1CT.cpu0.UTLB	TLB
ARMCortexA73x1CT.cpu0.l1dcache	PVCache
ARMCortexA73x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA73x1CT.cpu0.l1licache	PVCache
ARMCortexA73x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA73x1CT.ext_bus	PVBusLogger
ARMCortexA73x1CT.ext_bus.mapper	PVBusMapper
ARMCortexA73x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA73x1CT.l2_cache	PVCache
ARMCortexA73x1CT.l2_cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA73x1CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexA73x1CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexA73x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexA73x1CT contains the following CADI targets:

- ARM_Cortex-A73
- Cluster_ARM_Cortex-A73
- PVCache
- TlbCadi

About ARMCortexA73x1CT

- The following components also exist:
 - ARMCortexA73x2CT.
 - ARMCortexA73x3CT.
 - ARMCortexA73x4CT.
- The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).
- This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).
- If either L1 cache is stateful, then the L2 cache is stateful. This is controlled by the `dcache-state_modelled` and `icache-state_modelled` parameters.
- The `semihosting-cwd` parameter sets the CWD that is used for semihosting. The host operating system limits the maximum path length. The `semihosting-cwd` parameter does not provide any security. Software running on the model can access files outside this directory using relative paths containing `..` or using absolute paths.

- The system designer decides whether a debug APB ties the external debug view with other system views. In the model, use a `PVBusDecoder` to direct traffic to the correct port, `dev_debug_s` Or `memorymapped_debug_s`.
- ECC support is hardware-specific so is not modeled.

Ports for ARMCortexA73x1CT

Table 3-273: Ports

Name	Protocol	Type	Description
aa64naa32[4]	Signal	Slave	Register width after reset.
acp_s	PVBus	Slave	AXI ACP slave port.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastinner	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
cfgend[4]	Signal	Slave	This signal if for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[4]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clrexmonack	Signal	Master	Acknowledge handshake signal for the clrexmonreq signal
clrexmonreq	Signal	Slave	Signals the clearing of an external global exclusive monitor
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
CNTHPIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Timer signals to SOC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
commr[4]	Signal	Master	Receive portion of Data Transfer Register full.
commtx[4]	Signal	Master	Transmit portion of Data Transfer Register empty.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[4]	Signal	Slave	CPU power on reset.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross trigger matrix port.
dbgack[4]	Signal	Master	External debug interface.

Name	Protocol	Type	Description
dbgen[4]	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Processor powerup request.
dev_debug_s	PVBus	Slave	External debug interface.
edbgrq[4]	Signal	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
l2flushdone	Signal	Master	Flush of L2 memory system complete.
l2flushreq	Signal	Slave	Request flush of L2 memory system.
l2reset	Signal	Slave	Level2 reset.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[4]	Signal	Slave	External debug interface.
periphbase	Value_64	Slave	This port sets the base address of private peripheral region.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Debug reset.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt
reset[4]	Signal	Slave	Reset.
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sei[4]	Signal	Slave	Per core System Error physical pins.
smpen[4]	Signal	Master	This signals AMP or SMP mode for each core.
spiden[4]	Signal	Slave	External debug interface.
spniden[4]	Signal	Slave	External debug interface.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state
standbywfi12	Signal	Master	This signal indicates all cores and L2 are idle and in low power state.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.

Name	Protocol	Type	Description
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A73x1CT

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTINNER

Enable broadcasting of Inner Shareable transactions. The broadcastinner signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CCSIDR-L1D_override

If nonzero, override the value presented in CCSIDR for L1D (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L1I_override

If nonzero, override the value presented in CCSIDR for L1I (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L2_override

If nonzero, override the value presented in CCSIDR for L2 (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.SMPEN

Enable broadcast messages necessary for correct SMP operation at reset.

Type

bool

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

`icache-state_modelled`

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

`l2cache-hit_latency`

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l2cache-maintenance_latency`

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l2cache-miss_latency`

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l2cache-read_access_latency`

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `l2cache-read_latency` is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.26 ARMCortexA75CT

ARMCortexA75CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-274: IP revisions support

Revision	Quality level
r3p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA75CT

This model has the following Iris instances:

Table 3-275: ARMCortexA75CT Iris instances

InstanceName	ComponentName
ARMCortexA75CT	Cluster_ARM_Cortex-A75
ARMCortexA75CT.AMU	PVBusLogger
ARMCortexA75CT.AMU.mapper	PVBusMapper
ARMCortexA75CT.DAP	PVBusLogger
ARMCortexA75CT.DAP.mapper	PVBusMapper
ARMCortexA75CT.DSU	DSU
ARMCortexA75CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA75CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA75CT.DSU.shared_cache	PVCache
ARMCortexA75CT.DSU.shared_cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA75CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA75CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA75CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA75CT.MMAP	PVBusLogger
ARMCortexA75CT.MMAP.mapper	PVBusMapper
ARMCortexA75CT.cpu0	ARM_Cortex-A75
ARMCortexA75CT.cpu0.UTLB	TLB
ARMCortexA75CT.cpu0.dtlb	TlbCadi
ARMCortexA75CT.cpu0.l1dcache	PVCache
ARMCortexA75CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA75CT.cpu0.l1icache	PVCache
ARMCortexA75CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA75CT.cpu0.l2cache	PVCache
ARMCortexA75CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA75CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA75CT.ext_bus	PVBusLogger
ARMCortexA75CT.ext_bus.mapper	PVBusMapper
ARMCortexA75CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-276: ARMCortexA75CT MTI instances

InstanceName	ComponentName
ARMCortexA75CT	ARMv8Cluster
ARMCortexA75CT.AMU	PVBusLogger
ARMCortexA75CT.AMU.mapper	PVBusMapper
ARMCortexA75CT.DAP	PVBusLogger
ARMCortexA75CT.DAP.mapper	PVBusMapper
ARMCortexA75CT.DSU	DSU
ARMCortexA75CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA75CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA75CT.DSU.shared_cache	PVCache
ARMCortexA75CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA75CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA75CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA75CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA75CT.MMAP	PVBusLogger
ARMCortexA75CT.MMAP.mapper	PVBusMapper
ARMCortexA75CT.cpu0	ARM_Cortex-A75
ARMCortexA75CT.cpu0.UTLB	TLB

InstanceName	ComponentName
ARMCortexA75CT.cpu0.l1dcache	PVCache
ARMCortexA75CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA75CT.cpu0.l1icache	PVCache
ARMCortexA75CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA75CT.cpu0.l2cache	PVCache
ARMCortexA75CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA75CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA75CT.ext_bus	PVBusLogger
ARMCortexA75CT.ext_bus.mapper	PVBusMapper
ARMCortexA75CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA75CT contains the following CADI targets:

- ARM_Cortex-A75
- Cluster_ARM_Cortex-A75
- PVCache
- TlbCadi

About ARMCortexA75CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.

- Debug power management signals `DBGPWRDUP` and `DBGIRSTREQ` are not implemented, but `DBGPWRUPREQ` and `DBGNOPWRDWN` are implemented.
- Cache stashing capability.
- Dual ACE masters.

This model has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following models also exist, with a fixed number of cores per cluster:

- `ARMCortexA75x1CT`.
- `ARMCortexA75x2CT`.
- `ARMCortexA75x3CT`.
- `ARMCortexA75x4CT`.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA75CT

Table 3-277: Ports

Name	Protocol	Type	Description
<code>aa64naa32[4]</code>	Signal	Slave	Register width after reset.
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port
<code>AENDMP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[4]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>cfgte[4]</code>	Signal	Slave	This signal provides default exception handling state.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
<code>clusterid</code>	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
<code>clusterpmuirq</code>	Signal	Master	DynamlQ pmu irq
<code>CNTHPIRQ[4]</code>	Signal	Master	Timer signals to SOC.

Name	Protocol	Type	Description
CNTHVIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Timer signals to SOC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[4]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[4]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
edbgrq[4]	Signal	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[4]	PChannel	Slave	PChannels for cores
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.

Name	Protocol	Type	Description
rei[4]	Signal	Slave	Individual processor RAM Error Interrupt signal input.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[4]	Signal	Slave	Per core virtual System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtual FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtual IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A75CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CCSIDR-L1D_override

If nonzero, override the value presented in CCSIDR for L1D (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L1I_override

If nonzero, override the value presented in CCSIDR for L1I (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L2_override

If nonzero, override the value presented in CCSIDR for L2 (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.AA64nAA32

Register width configuration at reset. 0, AArch32. 1, AArch64.

Type

bool

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value

cpuX.semihosting-cwd

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.27 ARMCortexA76AECT

ARMCortexA76AECT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-278: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- tlbi_stall_enabled

Iris and MTI instances for ARMCortexA76AECT

This model has the following Iris instances:

Table 3-279: ARMCortexA76AECT Iris instances

InstanceName	ComponentName
ARMCortexA76AECT	Cluster_ARM_Cortex-A76AE

InstanceName	ComponentName
ARMCortexA76AECT.AMU	PVBusLogger
ARMCortexA76AECT.AMU.mapper	PVBusMapper
ARMCortexA76AECT.DAP	PVBusLogger
ARMCortexA76AECT.DAP.mapper	PVBusMapper
ARMCortexA76AECT.DSU	DSU
ARMCortexA76AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA76AECT.DSU.mpam_busslave	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache	PVCache
ARMCortexA76AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA76AECT.MMAP	PVBusLogger
ARMCortexA76AECT.MMAP.mapper	PVBusMapper
ARMCortexA76AECT.cpu0	ARM_Cortex-A76AE
ARMCortexA76AECT.cpu0.UTLB	TLB
ARMCortexA76AECT.cpu0.dtlb	TlbCadi
ARMCortexA76AECT.cpu0.l1dcache	PVCache
ARMCortexA76AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu0.l1licache	PVCache
ARMCortexA76AECT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu0.l2cache	PVCache
ARMCortexA76AECT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA76AECT.cpu1	ARM_Cortex-A76AE
ARMCortexA76AECT.cpu1.UTLB	TLB
ARMCortexA76AECT.cpu1.dtlb	TlbCadi
ARMCortexA76AECT.cpu1.l1dcache	PVCache
ARMCortexA76AECT.cpu1.l1dcache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu1.l1licache	PVCache
ARMCortexA76AECT.cpu1.l1licache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu1.l2cache	PVCache
ARMCortexA76AECT.cpu1.l2cache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu1.l2cache.upstream[1]	PVBusSlave
ARMCortexA76AECT.ext_bus	PVBusLogger
ARMCortexA76AECT.ext_bus.mapper	PVBusMapper
ARMCortexA76AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-280: ARMCortexA76AECT MTI instances

InstanceName	ComponentName
ARMCortexA76AECT	ARMv8Cluster
ARMCortexA76AECT.AMU	PVBusLogger
ARMCortexA76AECT.AMU.mapper	PVBusMapper
ARMCortexA76AECT.DAP	PVBusLogger
ARMCortexA76AECT.DAP.mapper	PVBusMapper
ARMCortexA76AECT.DSU	DSU
ARMCortexA76AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA76AECT.DSU.mpam_busslave	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache	PVCache
ARMCortexA76AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA76AECT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA76AECT.MMAP	PVBusLogger
ARMCortexA76AECT.MMAP.mapper	PVBusMapper
ARMCortexA76AECT.cpu0	ARM_Cortex-A76AE
ARMCortexA76AECT.cpu0.UTLB	TLB
ARMCortexA76AECT.cpu0.l1dcache	PVCache
ARMCortexA76AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu0.l1icache	PVCache
ARMCortexA76AECT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu0.l2cache	PVCache
ARMCortexA76AECT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA76AECT.cpu1	ARM_Cortex-A76AE
ARMCortexA76AECT.cpu1.UTLB	TLB
ARMCortexA76AECT.cpu1.l1dcache	PVCache
ARMCortexA76AECT.cpu1.l1dcache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu1.l1icache	PVCache
ARMCortexA76AECT.cpu1.l1icache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu1.l2cache	PVCache
ARMCortexA76AECT.cpu1.l2cache.upstream[0]	PVBusSlave
ARMCortexA76AECT.cpu1.l2cache.upstream[1]	PVBusSlave
ARMCortexA76AECT.ext_bus	PVBusLogger

InstanceName	ComponentName
ARMCortexA76AECT.ext_bus.mapper	PVBusMapper
ARMCortexA76AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA76AECT contains the following CADI targets:

- ARM_Cortex-A76AE
- Cluster_ARM_Cortex-A76AE
- PVCache
- TlbCadi

About ARMCortexA76AECT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals DBGPWRDUP and DBGRSTREQ are not implemented, but DBGPWRUPREQ and DBGNOPWRDWN are implemented.
- Cache stashing capability.
- Split/Lock is supported but with the limitations described in the AE-specific features implemented section.

This model has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following models also exist, with a fixed number of cores per cluster:

- ARMCortexA76AEx2CT.
- ARMCortexA76AEx4CT.
- ARMCortexA76AEx6CT.
- ARMCortexA76AEx8CT.

The per-core parameters are preceded by `cpu n .`, where n identifies the core (0-7).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

AE-specific features implemented

The only AE-specific features implemented in the model are Split-Lock and Lock-Step, with the following limitations:

- The implementation of Split-Lock and Lock-Step in the model is simply a halving of the number of cores available.
- Lock-step is implemented by a parameter that can be true or false for the entire execution only, rather than by a signal that can be enabled during execution.

As the model does not implement true Split-Lock functionality, the ports and parameters are contiguous. For example, for a 4-core CPU with Split-Lock enabled, `cpu0` and `cpu1` identify the available cores and associated ports, not `cpu0` and `cpu2`.

- Hybrid mode is not modeled in the DSU.

Ports for ARMCortexA76AECT

Table 3-281: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AENDMP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	<code>Value_64</code>	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[4]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgte[4]</code>	Signal	Slave	This signal provides default exception handling state.

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[4]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Timer signals to SOC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[4]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cpuporeset[4]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	These signals relate to core power down.
dbgprupreq[4]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[4]	PChannel	Slave	PChannels for cores
pmbirq[4]	Signal	Master	Interrupt signal from statistical profiling unit.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.

Name	Protocol	Type	Description
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[4]	Signal	Slave	Per core virtual System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A76 AECT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x2

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_lock_step

Whether the core is configured in Dual Core Lock Step mode (FEAT_DCLS)

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x100000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive (corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.28 **ARMCortexA76CT**

ARMCortexA76CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-282: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- tlbi_stall_enabled

Iris and MTI instances for ARM Cortex-A76CT

This model has the following Iris instances:

Table 3-283: ARM Cortex-A76CT Iris instances

InstanceName	ComponentName
ARM Cortex-A76CT	Cluster_ARM_Cortex-A76
ARM Cortex-A76CT.AMU	PVBusLogger
ARM Cortex-A76CT.AMU.mapper	PVBusMapper
ARM Cortex-A76CT.DAP	PVBusLogger
ARM Cortex-A76CT.DAP.mapper	PVBusMapper
ARM Cortex-A76CT.DSU	DSU
ARM Cortex-A76CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM Cortex-A76CT.DSU.mpam_busslave	PVBusSlave
ARM Cortex-A76CT.DSU.shared_cache	PVCache
ARM Cortex-A76CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM Cortex-A76CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARM Cortex-A76CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARM Cortex-A76CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARM Cortex-A76CT.MMAP	PVBusLogger
ARM Cortex-A76CT.MMAP.mapper	PVBusMapper
ARM Cortex-A76CT.cpu0	ARM_Cortex-A76
ARM Cortex-A76CT.cpu0.UTLB	TLB
ARM Cortex-A76CT.cpu0.dtlb	TlbCadi
ARM Cortex-A76CT.cpu0.l1dcache	PVCache
ARM Cortex-A76CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM Cortex-A76CT.cpu0.l1icache	PVCache
ARM Cortex-A76CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM Cortex-A76CT.cpu0.l2cache	PVCache
ARM Cortex-A76CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARM Cortex-A76CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARM Cortex-A76CT.ext_bus	PVBusLogger
ARM Cortex-A76CT.ext_bus.mapper	PVBusMapper
ARM Cortex-A76CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-284: ARM Cortex-A76CT MTI instances

InstanceName	ComponentName
ARM Cortex-A76CT	ARMv8Cluster
ARM Cortex-A76CT.AMU	PVBusLogger
ARM Cortex-A76CT.AMU.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA76CT.DAP	PVBusLogger
ARMCortexA76CT.DAP.mapper	PVBusMapper
ARMCortexA76CT.DSU	DSU
ARMCortexA76CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA76CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA76CT.DSU.shared_cache	PVCache
ARMCortexA76CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA76CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA76CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA76CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA76CT.MMAP	PVBusLogger
ARMCortexA76CT.MMAP.mapper	PVBusMapper
ARMCortexA76CT.cpu0	ARM_Cortex-A76
ARMCortexA76CT.cpu0.UTLB	TLB
ARMCortexA76CT.cpu0.l1dcache	PVCache
ARMCortexA76CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA76CT.cpu0.l1icache	PVCache
ARMCortexA76CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA76CT.cpu0.l2cache	PVCache
ARMCortexA76CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA76CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA76CT.ext_bus	PVBusLogger
ARMCortexA76CT.ext_bus.mapper	PVBusMapper
ARMCortexA76CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA76CT contains the following CADI targets:

- ARM_Cortex-A76
- Cluster_ARM_Cortex-A76
- PVCache
- TlbCadi

About ARMCortexA76CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.

- Per-core clock.
- `BROADCASTPERSIST` pin.

The model does not support the following features:

- `BROADCASTCACHEMAINTPOU` pin.
- `COREINSTRRET`, `COREINSTRRUN`, or `nPMBIRQ` signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals `DBGPWRDUP` and `DBGIRSTREQ` are not implemented, but `DBGPWRUPREQ` and `DBGNOPWRDWN` are implemented.
- Cache stashing capability.

This component has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following components also exist, with a fixed number of cores per cluster:

- `ARMCortexA76x1CT`.
- `ARMCortexA76x2CT`.
- `ARMCortexA76x3CT`.
- `ARMCortexA76x4CT`.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA76CT

Table 3-285: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AENDMP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	<code>Value_64</code>	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.

Name	Protocol	Type	Description
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[4]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[4]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[4]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Timer signals to SOC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[4]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cp15sdisable[4]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[4]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	These signals relate to core power down.
dbgpwrupreq[4]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.

Name	Protocol	Type	Description
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[4]	PChannel	Slave	PChannels for cores
pmbirq[4]	Signal	Master	Interrupt signal from statistical profiling unit.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[4]	Signal	Slave	Per core virtual System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA76CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`dcache-snoop_data_transfer_latency`

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`dcache-state_modelled`

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

`dcache-write_access_latency`

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size `dcache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`dcache-write_latency`

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. `dcache-write_access_latency` must be set to 0 for per-byte latencies to be applied. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x100000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.29 ARMCortexA77CT

ARMCortexA77CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-286: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `tlbi_stall_enabled`

Iris and MTI instances for ARMCortexA77CT

This model has the following Iris instances:

Table 3-287: ARMCortexA77CT Iris instances

InstanceName	ComponentName
ARMCortexA77CT	Cluster_ARM_Cortex-A77
ARMCortexA77CT.AMU	PVBusLogger
ARMCortexA77CT.AMU.mapper	PVBusMapper
ARMCortexA77CT.DAP	PVBusLogger
ARMCortexA77CT.DAP.mapper	PVBusMapper
ARMCortexA77CT.DSU	DSU
ARMCortexA77CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA77CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA77CT.DSU.shared_cache	PVCache
ARMCortexA77CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA77CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA77CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA77CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA77CT.MMAP	PVBusLogger
ARMCortexA77CT.MMAP.mapper	PVBusMapper
ARMCortexA77CT.cpu0	ARM_Cortex-A77
ARMCortexA77CT.cpu0.UTLB	TLB
ARMCortexA77CT.cpu0.dtlb	TlbCadi

InstanceName	ComponentName
ARMCortexA77CT.cpu0.l1dcache	PVCache
ARMCortexA77CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA77CT.cpu0.l1icache	PVCache
ARMCortexA77CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA77CT.cpu0.l2cache	PVCache
ARMCortexA77CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA77CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA77CT.ext_bus	PVBusLogger
ARMCortexA77CT.ext_bus.mapper	PVBusMapper
ARMCortexA77CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-288: ARMCortexA77CT MTI instances

InstanceName	ComponentName
ARMCortexA77CT	ARMv8Cluster
ARMCortexA77CT.AMU	PVBusLogger
ARMCortexA77CT.AMU.mapper	PVBusMapper
ARMCortexA77CT.DAP	PVBusLogger
ARMCortexA77CT.DAP.mapper	PVBusMapper
ARMCortexA77CT.DSU	DSU
ARMCortexA77CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA77CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA77CT.DSU.shared_cache	PVCache
ARMCortexA77CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA77CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA77CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA77CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA77CT.MMAP	PVBusLogger
ARMCortexA77CT.MMAP.mapper	PVBusMapper
ARMCortexA77CT.cpu0	ARM_Cortex-A77
ARMCortexA77CT.cpu0.UTLB	TLB
ARMCortexA77CT.cpu0.l1dcache	PVCache
ARMCortexA77CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA77CT.cpu0.l1icache	PVCache
ARMCortexA77CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA77CT.cpu0.l2cache	PVCache
ARMCortexA77CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA77CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA77CT.ext_bus	PVBusLogger

InstanceName	ComponentName
ARMCortexA77CT.ext_bus.mapper	PVBusMapper
ARMCortexA77CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA77CT contains the following CADI targets:

- ARM_Cortex-A77
- Cluster_ARM_Cortex-A77
- PVCache
- TlbCadi

About ARMCortexA77CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals DBGWRDUP and DBGRSTREQ are not implemented, but DBGWRUPREQ and DBGNOPWRDWN are implemented.
- Cache stashing capability.

This model has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following models also exist, with a fixed number of cores per cluster:

- ARMCortexA77x1CT.

- ARMCortexA77x2CT.
- ARMCortexA77x3CT.
- ARMCortexA77x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA77CT

Table 3-289: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AENDMP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[4]</code>	Signal	Slave	This signal if for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>cfgte[4]</code>	Signal	Slave	This signal provides default exception handling state.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
<code>clusterid</code>	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
<code>clusterpmuirq</code>	Signal	Master	DynamlQ pmu irq
<code>CNTHPIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>CNTHVIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>CNTPNSIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>CNTPSIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>cntvalueb</code>	CounterInterface	Slave	Interface to SoC level counter module.
<code>CNTVIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>commirq[4]</code>	Signal	Master	Interrupt signal from debug communications channel.
<code>core_clk_in[4]</code>	ClockSignal	Slave	The clock signal connected to the <code>core_clk_in</code> port is used to determine the rate at which each core executes instructions.

Name	Protocol	Type	Description
cpuporeset[4]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[4]	PChannel	Slave	PChannels for cores
pmbirq[4]	Signal	Master	Interrupt signal from statistical profiling unit.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[4]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.

Name	Protocol	Type	Description
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A77CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cpu_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-read_latency`

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-size`

L2 Cache size in bytes.

Type

int

Default value

0x80000

`cpuX.l2cache-snoop_data_transfer_latency`

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-snoop_issue_latency`

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x100000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.30 ARM Cortex A78AECT

ARM Cortex A78AECT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-290: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- tlbi_stall_enabled

Iris and MTI instances for ARM Cortex A78AECT

This model has the following Iris instances:

Table 3-291: ARM Cortex A78AECT Iris instances

InstanceName	ComponentName
ARM Cortex A78AECT	Cluster_ARM_Cortex-A78AE
ARM Cortex A78AECT.AMU	PVBusLogger
ARM Cortex A78AECT.AMU.mapper	PVBusMapper
ARM Cortex A78AECT.DAP	PVBusLogger
ARM Cortex A78AECT.DAP.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA78AECT.DSU	DSU
ARMCortexA78AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA78AECT.DSU.mpam_busslave	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache	PVCache
ARMCortexA78AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA78AECT.MMAP	PVBusLogger
ARMCortexA78AECT.MMAP.mapper	PVBusMapper
ARMCortexA78AECT.cpu0	ARM_Cortex-A78AE
ARMCortexA78AECT.cpu0.UTLB	TLB
ARMCortexA78AECT.cpu0.dtlb	TlbCadi
ARMCortexA78AECT.cpu0.l1dcache	PVCache
ARMCortexA78AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu0.l1licache	PVCache
ARMCortexA78AECT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu0.l2cache	PVCache
ARMCortexA78AECT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA78AECT.cpu1	ARM_Cortex-A78AE
ARMCortexA78AECT.cpu1.UTLB	TLB
ARMCortexA78AECT.cpu1.dtlb	TlbCadi
ARMCortexA78AECT.cpu1.l1dcache	PVCache
ARMCortexA78AECT.cpu1.l1dcache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu1.l1licache	PVCache
ARMCortexA78AECT.cpu1.l1licache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu1.l2cache	PVCache
ARMCortexA78AECT.cpu1.l2cache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu1.l2cache.upstream[1]	PVBusSlave
ARMCortexA78AECT.ext_bus	PVBusLogger
ARMCortexA78AECT.ext_bus.mapper	PVBusMapper
ARMCortexA78AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-292: ARMCortexA78AECT MTI instances

InstanceName	ComponentName
ARMCortexA78AECT	ARMv8Cluster
ARMCortexA78AECT.AMU	PVBusLogger
ARMCortexA78AECT.AMU.mapper	PVBusMapper
ARMCortexA78AECT.DAP	PVBusLogger
ARMCortexA78AECT.DAP.mapper	PVBusMapper
ARMCortexA78AECT.DSU	DSU
ARMCortexA78AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA78AECT.DSU.mpam_busslave	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache	PVCache
ARMCortexA78AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA78AECT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA78AECT.MMAP	PVBusLogger
ARMCortexA78AECT.MMAP.mapper	PVBusMapper
ARMCortexA78AECT.cpu0	ARM_Cortex-A78AE
ARMCortexA78AECT.cpu0.UTLB	TLB
ARMCortexA78AECT.cpu0.l1dcache	PVCache
ARMCortexA78AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu0.l1licache	PVCache
ARMCortexA78AECT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu0.l2cache	PVCache
ARMCortexA78AECT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA78AECT.cpu1	ARM_Cortex-A78AE
ARMCortexA78AECT.cpu1.UTLB	TLB
ARMCortexA78AECT.cpu1.l1dcache	PVCache
ARMCortexA78AECT.cpu1.l1dcache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu1.l1licache	PVCache
ARMCortexA78AECT.cpu1.l1licache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu1.l2cache	PVCache
ARMCortexA78AECT.cpu1.l2cache.upstream[0]	PVBusSlave
ARMCortexA78AECT.cpu1.l2cache.upstream[1]	PVBusSlave
ARMCortexA78AECT.ext_bus	PVBusLogger
ARMCortexA78AECT.ext_bus.mapper	PVBusMapper
ARMCortexA78AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA78AECT contains the following CADI targets:

- ARM_Cortex-A78AE
- Cluster_ARM_Cortex-A78AE
- PVCache
- TlbCadi

About ARMCortexA78AECT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals DBGPWRDUP and DBGRSTREQ are not implemented, but DBGPWRUPREQ and DBGNOFWRDWN are implemented.
- Cache stashing capability.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.
- Split/Lock is supported but with the limitations described in the AE-specific features implemented section.

This model has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following models also exist, with a fixed number of cores per cluster:

- ARMCortexA78AEx2CT.
- ARMCortexA78AEx4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

AE-specific features implemented

The only AE-specific features implemented in the model are Split-Lock and Lock-Step, with the following limitations:

- The implementation of Split-Lock and Lock-Step in the model is simply a halving of the number of cores available.
- Lock-step is implemented by a parameter that can be true or false for the entire execution only, rather than by a signal that can be enabled during execution.

As the model does not implement true Split-Lock functionality, the ports and parameters are contiguous. For example, for a 4-core CPU with Split-Lock enabled, `cpu0` and `cpu1` identify the available cores and associated ports, not `cpu0` and `cpu2`.

- Hybrid mode is not modeled in the DSU.

Ports for ARMCortexA78AECT

Table 3-293: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AENDMP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[4]</code>	Signal	Slave	This signal if for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>cfgte[4]</code>	Signal	Slave	This signal provides default exception handling state.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
<code>clusterid</code>	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.

Name	Protocol	Type	Description
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[4]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Timer signals to SOC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[4]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cpuporeset[4]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[4]	PChannel	Slave	PChannels for cores
pmbirq[4]	Signal	Master	Interrupt signal from statistical profiling unit.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.

Name	Protocol	Type	Description
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[4]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA78AECT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x2

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_lock_step

Whether the core is configured in Dual Core Lock Step mode (FEAT_DCLS)

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x0

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or

miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x100000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive (corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.31 ARM Cortex A78CCT

ARM Cortex A78CCT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-294: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `tlbi_stall_enabled`

Iris and MTI instances for ARM Cortex-A78CCT

This model has the following Iris instances:

Table 3-295: ARM Cortex-A78CCT Iris instances

InstanceName	ComponentName
ARM Cortex-A78CCT	Cluster_ARM_Cortex-A78C
ARM Cortex-A78CCT.AMU	PVBusLogger
ARM Cortex-A78CCT.AMU.mapper	PVBusMapper
ARM Cortex-A78CCT.DAP	PVBusLogger
ARM Cortex-A78CCT.DAP.mapper	PVBusMapper
ARM Cortex-A78CCT.DSU	DSU
ARM Cortex-A78CCT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM Cortex-A78CCT.DSU.mpam_busslave	PVBusSlave
ARM Cortex-A78CCT.DSU.shared_cache	PVCache
ARM Cortex-A78CCT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM Cortex-A78CCT.DSU.shared_cache.upstream[1]	PVBusSlave
ARM Cortex-A78CCT.DSU.shared_cache.upstream[2]	PVBusSlave
ARM Cortex-A78CCT.DSU.shared_cache.upstream[3]	PVBusSlave
ARM Cortex-A78CCT.MMAP	PVBusLogger
ARM Cortex-A78CCT.MMAP.mapper	PVBusMapper
ARM Cortex-A78CCT.cpu0	ARM_Cortex-A78C
ARM Cortex-A78CCT.cpu0.UTLB	TLB
ARM Cortex-A78CCT.cpu0.dtlb	TlbCadi
ARM Cortex-A78CCT.cpu0.l1dcache	PVCache
ARM Cortex-A78CCT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM Cortex-A78CCT.cpu0.l1icache	PVCache
ARM Cortex-A78CCT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM Cortex-A78CCT.cpu0.l2cache	PVCache
ARM Cortex-A78CCT.cpu0.l2cache.upstream[0]	PVBusSlave
ARM Cortex-A78CCT.cpu0.l2cache.upstream[1]	PVBusSlave
ARM Cortex-A78CCT.ext_bus	PVBusLogger

InstanceName	ComponentName
ARMCortexA78CCT.ext_bus.mapper	PVBusMapper
ARMCortexA78CCT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-296: ARMCortexA78CCT MTI instances

InstanceName	ComponentName
ARMCortexA78CCT	ARMv8Cluster
ARMCortexA78CCT.AMU	PVBusLogger
ARMCortexA78CCT.AMU.mapper	PVBusMapper
ARMCortexA78CCT.DAP	PVBusLogger
ARMCortexA78CCT.DAP.mapper	PVBusMapper
ARMCortexA78CCT.DSU	DSU
ARMCortexA78CCT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA78CCT.DSU.mpam_busslave	PVBusSlave
ARMCortexA78CCT.DSU.shared_cache	PVCache
ARMCortexA78CCT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA78CCT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA78CCT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA78CCT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA78CCT.MMAP	PVBusLogger
ARMCortexA78CCT.MMAP.mapper	PVBusMapper
ARMCortexA78CCT.cpu0	ARM_Cortex-A78C
ARMCortexA78CCT.cpu0.UTLB	TLB
ARMCortexA78CCT.cpu0.l1dcache	PVCache
ARMCortexA78CCT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA78CCT.cpu0.l1icache	PVCache
ARMCortexA78CCT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA78CCT.cpu0.l2cache	PVCache
ARMCortexA78CCT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA78CCT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA78CCT.ext_bus	PVBusLogger
ARMCortexA78CCT.ext_bus.mapper	PVBusMapper
ARMCortexA78CCT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA78CCT contains the following CADI targets:

- ARM_Cortex-A78C
- Cluster_ARM_Cortex-A78C
- PVCache

- TlbCadi

About ARMCortexA78CCT

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals DBGPWDUP and DBGSTREQ are not implemented, but DBGWRUPREQ and DBGNOPWRDN are implemented.
- Cache stashing capability.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.

This model has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following models also exist, with a fixed number of cores per cluster:

- ARMCortexA78x1CCT.
- ARMCortexA78x2CCT.
- ARMCortexA78x4CCT.
- ARMCortexA78x6CCT.
- ARMCortexA78x8CCT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-7).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (crypto.dll or crypto.so) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA78CCT

Table 3-297: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
AENDMP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTARTMP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[8]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[8]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[8]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[8]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[8]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[8]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[8]	Signal	Master	Timer signals to SOC.
commirq[8]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[8]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cpuporeset[8]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[8]	Signal	Slave	Disable cryptography extensions after reset.
cti[8]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[8]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.

Name	Protocol	Type	Description
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[8]	Signal	Master	No power-down request.
dbgpwrupreq[8]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[8]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[8]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[8]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[8]	PChannel	Slave	PChannels for cores
pmbirq[8]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[8]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[8]	Signal	Slave	Per core RAM Error Interrupt.
reset[8]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[8]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[8]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynaMIQ.
ticks[8]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[8]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.

Name	Protocol	Type	Description
vfiq[8]	Signal	Slave	Virtualised FIQ.
vinithi[8]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[8]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[8]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A78CCT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x0

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x100000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.32 ARMCortexA78CT

ARMCortexA78CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-298: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `tlbi_stall_enabled`

Iris and MTI instances for ARMCortexA78CT

This model has the following Iris instances:

Table 3-299: ARMCortexA78CT Iris instances

InstanceName	ComponentName
ARMCortexA78CT	Cluster_ARM_Cortex-A78
ARMCortexA78CT.AMU	PVBusLogger
ARMCortexA78CT.AMU.mapper	PVBusMapper
ARMCortexA78CT.DAP	PVBusLogger
ARMCortexA78CT.DAP.mapper	PVBusMapper
ARMCortexA78CT.DSU	DSU
ARMCortexA78CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA78CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA78CT.DSU.shared_cache	PVCache
ARMCortexA78CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA78CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA78CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA78CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA78CT.MMAP	PVBusLogger
ARMCortexA78CT.MMAP.mapper	PVBusMapper
ARMCortexA78CT.cpu0	ARM_Cortex-A78
ARMCortexA78CT.cpu0.UTLB	TLB
ARMCortexA78CT.cpu0.dtlb	TlbCadi

InstanceName	ComponentName
ARMCortexA78CT.cpu0.l1dcache	PVCache
ARMCortexA78CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA78CT.cpu0.l1icache	PVCache
ARMCortexA78CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA78CT.cpu0.l2cache	PVCache
ARMCortexA78CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA78CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA78CT.ext_bus	PVBusLogger
ARMCortexA78CT.ext_bus.mapper	PVBusMapper
ARMCortexA78CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-300: ARMCortexA78CT MTI instances

InstanceName	ComponentName
ARMCortexA78CT	ARMv8Cluster
ARMCortexA78CT.AMU	PVBusLogger
ARMCortexA78CT.AMU.mapper	PVBusMapper
ARMCortexA78CT.DAP	PVBusLogger
ARMCortexA78CT.DAP.mapper	PVBusMapper
ARMCortexA78CT.DSU	DSU
ARMCortexA78CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA78CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA78CT.DSU.shared_cache	PVCache
ARMCortexA78CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA78CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA78CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA78CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA78CT.MMAP	PVBusLogger
ARMCortexA78CT.MMAP.mapper	PVBusMapper
ARMCortexA78CT.cpu0	ARM_Cortex-A78
ARMCortexA78CT.cpu0.UTLB	TLB
ARMCortexA78CT.cpu0.l1dcache	PVCache
ARMCortexA78CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA78CT.cpu0.l1icache	PVCache
ARMCortexA78CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA78CT.cpu0.l2cache	PVCache
ARMCortexA78CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA78CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA78CT.ext_bus	PVBusLogger

InstanceName	ComponentName
ARMCortexA78CT.ext_bus.mapper	PVBusMapper
ARMCortexA78CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA78CT contains the following CADI targets:

- ARM_Cortex-A78
- Cluster_ARM_Cortex-A78
- PVCache
- TlbCadi

About ARMCortexA78CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals DBGPRWDUP and DBGRSTREQ are not implemented, but DBGPRWRUPREQ and DBGNOFWRDWN are implemented.
- Cache stashing capability.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.

This component has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following components also exist, with a fixed number of cores per cluster:

- ARMCortexA78x1CT.
- ARMCortexA78x2CT.
- ARMCortexA78x3CT.
- ARMCortexA78x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA78CT

Table 3-301: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AENDMP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[4]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>cfgte[4]</code>	Signal	Slave	This signal provides default exception handling state.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
<code>clusterid</code>	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
<code>clusterpmuirq</code>	Signal	Master	DynamlQ pmu irq
<code>CNTHPIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>CNTHVIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>CNTPNSIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>CNTPSIRQ[4]</code>	Signal	Master	Timer signals to SOC.
<code>cntvalueb</code>	CounterInterface	Slave	Interface to SoC level counter module.
<code>CNTVIRQ[4]</code>	Signal	Master	Timer signals to SOC.

Name	Protocol	Type	Description
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[4]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cpuporeset[4]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[4]	PChannel	Slave	PChannels for cores
pmbirq[4]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain

Name	Protocol	Type	Description
sei[4]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A78CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cpu_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value

cpuX.semihosting-cwd

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x0

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x100000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive (corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.33 ARM Cortex A510CT

ARM Cortex A510CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-302: IP revisions support

Revision	Quality level
r0p0	Full support
r1p3	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Table 3-303: Model quality changes

From	To
r0p0=rel	r0p0=rel
r1p0=rel	r1p3=rel

Parameters added:

- `mpmm_accumulator_multiplier`
- `tlbi_stall_enabled`

Iris and MTI instances for ARM Cortex A510CT

This model has the following Iris instances:

Table 3-304: ARM Cortex A510CT Iris instances

InstanceName	ComponentName
ARM Cortex A510CT	Cluster_ARM_Cortex-A510
ARM Cortex A510CT.AMU	PVBusLogger
ARM Cortex A510CT.AMU.mapper	PVBusMapper
ARM Cortex A510CT.DAP	PVBusLogger
ARM Cortex A510CT.DAP.mapper	PVBusMapper
ARM Cortex A510CT.DSU	DSU

InstanceName	ComponentName
ARMCortexA510CT.DSU.PPU_cluster	PPUv1
ARMCortexA510CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA510CT.DSU.PPU_core0	PPUv1
ARMCortexA510CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA510CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA510CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA510CT.DSU.shared_cache	PVCache
ARMCortexA510CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA510CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA510CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA510CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA510CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA510CT.MMAP	PVBusLogger
ARMCortexA510CT.MMAP.mapper	PVBusMapper
ARMCortexA510CT.cpu0	ARM_Cortex-A510
ARMCortexA510CT.cpu0.UTLB	TLB
ARMCortexA510CT.cpu0.dtlb	TlbCadi
ARMCortexA510CT.cpu0.l1dcache	PVCache
ARMCortexA510CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT.cpu0.l1icache	PVCache
ARMCortexA510CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT.cpu0.l2cache	PVCache
ARMCortexA510CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT.ext_bus	PVBusLogger
ARMCortexA510CT.ext_bus.mapper	PVBusMapper
ARMCortexA510CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-305: ARMCortexA510CT MTI instances

InstanceName	ComponentName
ARMCortexA510CT	ARMv8Cluster
ARMCortexA510CT.AMU	PVBusLogger
ARMCortexA510CT.AMU.mapper	PVBusMapper
ARMCortexA510CT.DAP	PVBusLogger
ARMCortexA510CT.DAP.mapper	PVBusMapper
ARMCortexA510CT.DSU	DSU
ARMCortexA510CT.DSU.PPU_cluster	PPUv1
ARMCortexA510CT.DSU.PPU_cluster.busslave	PVBusSlave

InstanceName	ComponentName
ARMCortexA510CT.DSU.PPU_core0	PPUv1
ARMCortexA510CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA510CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA510CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA510CT.DSU.shared_cache	PVCache
ARMCortexA510CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA510CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA510CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA510CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA510CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA510CT.MMAP	PVBusLogger
ARMCortexA510CT.MMAP.mapper	PVBusMapper
ARMCortexA510CT.cpu0	ARM_Cortex-A510
ARMCortexA510CT.cpu0.UTLB	TLB
ARMCortexA510CT.cpu0.l1dcache	PVCache
ARMCortexA510CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT.cpu0.l1icache	PVCache
ARMCortexA510CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT.cpu0.l2cache	PVCache
ARMCortexA510CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT.ext_bus	PVBusLogger
ARMCortexA510CT.ext_bus.mapper	PVBusMapper
ARMCortexA510CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA510CT contains the following CADI targets:

- ARM_Cortex-A510
- Cluster_ARM_Cortex-A510
- PVCache
- TlbCadi

About ARMCortexA510CT

The model supports the following features:

- DynamIQ™ Shared Unit-110 (DSU-110) system registers.
- L2 cache is supported at the per-core level only and there is no implementation of Core-Complex with shared L2 cache yet.
- PChannel for the cluster and for each core.
- BROADCASTPERSIST pin.

- Optional peripheral port.
- L3Cache partition.
- Per-core clock.
- Utility bus.
- Revision R1 is the default configuration, with 32-bit support at ELO. R1 supports both configurations of ELO, with or without A32 support. For 64-bit only mode, set parameter `max_32bit_el=-1`.
- To configure revision R0, set parameter `revision_number=0`.

Support for the following features is planned for a future release:

- Transporting architectural metadata tags over TLM2 (OSCI) and AMBA-PV (the Arm TLM2 extensions) bus protocols.
- `BROADCASTCACHEMAINTPOU` pin.
- `COREINSTRRET`, `COREINSTRRUN`, and `nPMBIRQ` signals.

The following features will not be implemented:

- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- Error correction or detection features.
- Self-test features (MBIST).
- Snoop filtering.
- Latency configuration.
- Cache stashing capability.

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA510CT

Table 3-306: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AENDOMP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND1MP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND2MP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).

Name	Protocol	Type	Description
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[12]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[12]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[12]	Signal	Master	Timer signals to SOC
CNTHVIRQ[12]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[12]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[12]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[12]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[12]	Signal	Master	Timer signals to SOC.
commirq[12]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[12]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[12]	PChannel	Master	Core PCSM signals

Name	Protocol	Type	Description
core_powerdown_out[12]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[12]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[12]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[12]	Signal	Slave	Disable cryptography extensions after reset.
cti[12]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[12]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[12]	Signal	Master	No power-down request.
dbgprupreq[12]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[12]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[12]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[12]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmuirq[12]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[12]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[12]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.

Name	Protocol	Type	Description
rei[12]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[12]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[12]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[12]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[12]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[12]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[12]	Signal	Slave	Virtualised FIQ.
virq[12]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[12]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A510CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-read_bus_width_in_bytes`

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

`cpuX.l2cache-read_latency`

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-size`

L2 Cache size in bytes.

Type

int

Default value

0x80000

`cpuX.l2cache-snoop_data_transfer_latency`

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-snoop_issue_latency`

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size `dcache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCSRSTA_FORCED_EXCEP**

TRCSRSTA value for a forcibly traced exception

Type

bool

Default value

0x0

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

max_32bit_el

Maximum exception level supporting AArch32 modes. -1: No Support for A32 at any EL, 0: ELO supports A32. This parameter is ignored in Rev0 i.e. when revision_number = 0

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

patch_level

Patch level of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Revision field in MIDR/MIDR_EL1. Corresponds to the patch number Y in rXpY.

Type

int

Default value

0x3

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x1

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

- tlbi_stall_enabled**
If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.
Type
bool
Default value
0x0
- treat_PAC_as_NOP**
Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.
Type
bool
Default value
0x0
- walk_cache_latency**
Walk cache latency for TA (Timing Annotation), expressed in simulation ticks
Type
int
Default value
0x0

3.5.34 ARMCortexA510CT_CortexA710CT

ARMCortexA510CT_CortexA710CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-307: IP revisions support

Revision	Quality level
CortexA510 r0p0	Full support
CortexA510 r1p3	Full support
CortexA710 r2p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Table 3-308: Model quality changes

From	To
rel	CortexA510 r0p0=rel CortexA510 r1p3=rel CortexA710 r2p0=rel

Table 3-309: IP revision changes

From	To
CortexA510 r0p0	CortexA510 r0p0
CortexA510 r1p3	CortexA510 r1p3
CortexA710 r2p0	CortexA710 r2p0

Ports added:

- cluster_pcsn_pchannel
- core_pcsn_pchannel

Parameters added:

- mpmmm_accumulator_multiplier
- subcluster0.tlbi_stall_enabled
- subcluster1.tlbi_stall_enabled

Iris and MTI instances for ARMCortexA510CT_CortexA710CT

This model has the following Iris instances:

Table 3-310: ARMCortexA510CT_CortexA710CT Iris instances

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT	Cluster_ARM_Cortex510_CortexA710_Heterogeneous
ARMCortexA510CT_CortexA710CT.AMU	PVBusLogger
ARMCortexA510CT_CortexA710CT.AMU.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT.DAP	PVBusLogger
ARMCortexA510CT_CortexA710CT.DAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT.DSU	DSU
ARMCortexA510CT_CortexA710CT.DSU.PPU_cluster	PPUv1
ARMCortexA510CT_CortexA710CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.PPU_core0	PPUv1
ARMCortexA510CT_CortexA710CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.PPU_core1	PPUv1
ARMCortexA510CT_CortexA710CT.DSU.PPU_core1.busslave	PVBusSlave

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache	PVCache
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.MMAP	PVBusLogger
ARMCortexA510CT_CortexA710CT.MMAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT.ext_bus	PVBusLogger
ARMCortexA510CT_CortexA710CT.ext_bus.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA510CT_CortexA710CT.subcluster0	Subcluster_ARM_Cortex-A510
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0	ARM_Cortex-A510
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster1	Subcluster_ARM_Cortex-A710
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0	ARM_Cortex-A710
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l1licache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster1.cpu1.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-311: ARMCortexA510CT_CortexA710CT MTI instances

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT	ARMv8Cluster
ARMCortexA510CT_CortexA710CT.AMU	PVBusLogger
ARMCortexA510CT_CortexA710CT.AMU.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT.DAP	PVBusLogger
ARMCortexA510CT_CortexA710CT.DAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT.DSU	DSU
ARMCortexA510CT_CortexA710CT.DSU.PPU_cluster	PPUv1
ARMCortexA510CT_CortexA710CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.PPU_core0	PPUv1
ARMCortexA510CT_CortexA710CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.PPU_core1	PPUv1
ARMCortexA510CT_CortexA710CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache	PVCache
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA510CT_CortexA710CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.MMAP	PVBusLogger
ARMCortexA510CT_CortexA710CT.MMAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT.ext_bus	PVBusLogger
ARMCortexA510CT_CortexA710CT.ext_bus.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0	ARM_Cortex-A510
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0	ARM_Cortex-A710
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.TLB	TLB
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA510CT_CortexA710CT contains the following CADI targets:

- ARM_Cortex-A510
- ARM_Cortex-A710
- Cluster_ARM_Cortex510_CortexA710_Heterogeneous
- PVCache
- Subcluster_ARM_Cortex-A510
- Subcluster_ARM_Cortex-A710
- TlbCadi

About ARMCortexA510CT_CortexA710CT

The number of cores in each subcluster is configurable using the following parameters:

subcluster0.NUM_CORES

Possible values are 1-11 (ARMCortexA510CT).

subcluster1.NUM_CORES

Possible values are 1-11 (ARMCortexA710CT).

The total number of cores in the cluster cannot exceed 12.

Port arrays are expanded to the width needed for the maximum number of cores in each subcluster. Use the following port array indexes:

- `<port_name>[0-10]` for cores in subcluster0.
- `<port_name>[11-21]` for cores in subcluster1.



All instances in the Master cross trigger matrix port array `cti[22]` must be connected, regardless of the `NUM_CORES` value used.

Core-specific parameters have the following prefixes:

- `subcluster0.cpu0` to `subcluster0.cpu10` identify cores in `subcluster0`.
- `subcluster1.cpu0` to `subcluster1.cpu10` identify cores in `subcluster1`.

For information about the cores in this model, see:

- [3.5.33 ARMCortexA510CT](#) on page 1401.
- [3.5.44 ARMCortexA710CT](#) on page 1964.

Ports for ARMCortexA510CT_CortexA710CT

Table 3-312: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AEND0MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND1MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND2MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND3MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTART0MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART1MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART2MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART3MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[22]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.

Name	Protocol	Type	Description
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[22]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[22]	Signal	Master	Timer signals to SOC
CNTHVIRQ[22]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[22]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[22]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[22]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[22]	Signal	Master	Timer signals to SOC.
commirq[22]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[22]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[22]	PChannel	Master	Core PCSM signals
core_powerdown_out[22]	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the core power info
coreerrirq[22]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[22]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[22]	Signal	Slave	Disable cryptography extensions after reset.
cti[22]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[22]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[22]	Signal	Master	No power-down request.
dbgpwrupreq[22]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.

Name	Protocol	Type	Description
fiq[22]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[22]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[22]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmuirq[22]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[22]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[22]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[22]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[22]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[22]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[22]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[22]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[22]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[22]	Signal	Slave	Virtualised FIQ.
virq[22]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).

Name	Protocol	Type	Description
vsei[22]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM CortexA510CT_CortexA710CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x0

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

subcluster0.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 12

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster0.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster0.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster0.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster0.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster0.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster0.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster0.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster0.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster0.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster0.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster0.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster0.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster0.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster0.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.max_32bit_el

Maximum exception level supporting AArch32 modes. -1: No Support for A32 at any EL, 0: ELO supports A32. This parameter is ignored in Rev0 i.e. when revision_number = 0

Type

int

Default value

0x0

subcluster0.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

subcluster0.revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x1

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 12

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size `dcache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. `dcache-write_access_latency` must be set to 0 for per-byte latencies to be applied. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x20

subcluster1.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster1.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster1.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x1

subcluster1.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster1.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster1.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster1.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster1.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x2

subcluster1.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster1.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster1.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster1.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster1.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster1.ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

subcluster1.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster1.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster1.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).

Type

int

Default value

0x2

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.35 ARMCortexA510CT_CortexA710CT_CortexX2CT

ARMCortexA510CT_CortexA710CT_CortexX2CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-313: IP revisions support

Revision	Quality level
CortexA510 r0p0	Full support
CortexA510 r1p3	Full support

Revision	Quality level
CortexA710 r2p0	Full support
CortexX2 r2p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Table 3-314: Model quality changes

From	To
rel	CortexA510 r0p0=rel CortexA510 r1p3=rel CortexA710 r2p0=rel CortexX2 r2p0=rel

Table 3-315: IP revision changes

From	To
CortexA510 r0p0	CortexA510 r0p0
CortexA510 r1p3	CortexA510 r1p3
CortexA710 r2p0	CortexA710 r2p0
CortexX2 r2p0	CortexX2 r2p0

Ports added:

- cluster_pcs_m_pchannel
- core_pcs_m_pchannel

Ports removed:

- pm_birq

Parameters added:

- mpm_accumulator_multiplier
- subcluster0.tlbi_stall_enabled
- subcluster1.tlbi_stall_enabled
- subcluster2.tlbi_stall_enabled

Iris and MTI instances for ARMCortexA510CT_CortexA710CT_CortexX2CT

This model has the following Iris instances:

Table 3-316: ARMCortexA510CT_CortexA710CT_CortexX2CT Iris instances

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX2CT	Cluster_ARM_CortexA510_CortexA710_CortexX2_Heterogeneous
ARMCortexA510CT_CortexA710CT_CortexX2CT.AMU	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX2CT.AMU.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX2CT.DAP	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX2CT.DAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU	DSU
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_cluster	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core0	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core1	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core2	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core2.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[7]	PVBusSlave

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.MMAP	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX2CT.MMAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX2CT.ext_bus	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX2CT.ext_bus.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX2CT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0	Subcluster_ARM_Cortex-A510
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0	ARM_Cortex-A510
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1	Subcluster_ARM_Cortex-A710
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0	ARM_Cortex-A710
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu1.dtlb	TlbCadi
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2	Subcluster_ARM_Cortex-X2
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0	ARM_Cortex-X2
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu2.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-317: ARMCortexA510CT_CortexA710CT_CortexX2CT MTI instances

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX2CT	ARMv8Cluster
ARMCortexA510CT_CortexA710CT_CortexX2CT.AMU	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX2CT.AMU.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX2CT.DAP	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX2CT.DAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU	DSU
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_cluster	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core0	PPUv1

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core1	PPUV1
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core2	PPUV1
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.PPU_core2.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.shared_cache.upstream[7]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.MMAP	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX2CT.MMAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX2CT.ext_bus	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX2CT.ext_bus.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX2CT.gic_cpuif_decoder_cluster	GIcV3CPUInterfaceDecoder
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0	ARM_Cortex-A510
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0	ARM_Cortex-A710
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l1licache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0	ARM_Cortex-X2
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX2CT.subcluster2.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA510CT_CortexA710CT_CortexX2CT contains the following CADI targets:

- ARM_Cortex-A510
- ARM_Cortex-A710
- ARM_Cortex-X2
- Cluster_ARM_CortexA510_CortexA710_CortexX2_Heterogeneous
- PVCache
- Subcluster_ARM_Cortex-A510
- Subcluster_ARM_Cortex-A710
- Subcluster_ARM_Cortex-X2
- TlbCadi

About ARMCortexA510CT_CortexA710CT_CortexX2CT

The number of cores in each subcluster is configurable using the following parameters:

subcluster0.NUM_CORES

Possible values are 1-10 (ARMCortexA510CT).

subcluster1.NUM_CORES

Possible values are 1-10 (ARMCortexA710CT).

subcluster2.NUM_CORES

Possible values are 1-10 (ARMCortexX2CT).

The total number of cores in the cluster cannot exceed 12.

Port arrays are expanded to the width needed for the maximum number of cores in each subcluster. Use the following port array indexes:

- `<port_name>[0-9]` for cores in subcluster0.
- `<port_name>[10-19]` for cores in subcluster1.
- `<port_name>[20-29]` for cores in subcluster2.



All instances in the Master cross trigger matrix port array, `cti[30]` must be connected, regardless of the `NUM_CORES` value used.

Core-specific parameters have the following prefixes:

- `subcluster0.cpu0` to `subcluster0.cpu9` identify cores in `subcluster0`.
- `subcluster1.cpu0` to `subcluster1.cpu9` identify cores in `subcluster1`.
- `subcluster2.cpu0` to `subcluster2.cpu9` identify cores in `subcluster2`.

For information about the cores in this model, see:

- [3.5.33 ARMCortexA510CT](#) on page 1401.
- [3.5.44 ARMCortexA710CT](#) on page 1964.
- [3.5.71 ARMCortexX2CT](#) on page 2493.

Ports for ARMCortexA510CT_CortexA710CT_CortexX2CT

Table 3-318: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AEND0MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND1MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND2MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND3MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTART0MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART1MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART2MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART3MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[30]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[30]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[30]	Signal	Master	Timer signals to SOC
CNTHVIRQ[30]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[30]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[30]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[30]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[30]	Signal	Master	Timer signals to SOC.
commirq[30]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[30]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[30]	PChannel	Master	Core PCSM signals
core_powerdown_out[30]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[30]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[30]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[30]	Signal	Slave	Disable cryptography extensions after reset.
cti[30]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[30]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[30]	Signal	Master	No power-down request.
dbgprupreq[30]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.

Name	Protocol	Type	Description
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[30]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[30]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[30]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmuirq[30]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[30]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[30]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[30]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[30]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[30]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[30]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[30]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[30]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[30]	Signal	Slave	Virtualised FIQ.
virq[30]	Signal	Slave	Virtualised IRQ.

Name	Protocol	Type	Description
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[30]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A510CT_Cortex A710CT_Cortex X2CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAJINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x0

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases

where execution of quantums in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

subcluster0.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 12

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.etc.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster0.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster0.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster0.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster0.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster0.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster0.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster0.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster0.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster0.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster0.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster0.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster0.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster0.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster0.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.max_32bit_el

Maximum exception level supporting AArch32 modes. -1: No Support for A32 at any EL, 0: ELO supports A32. This parameter is ignored in Rev0 i.e. when revision_number = 0

Type

int

Default value

0x0

subcluster0.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

subcluster0.revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x1

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 12

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size `dcache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. `dcache-write_access_latency` must be set to 0 for per-byte latencies to be applied. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x20

subcluster1.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster1.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster1.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x1

subcluster1.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster1.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster1.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster1.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster1.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x2

subcluster1.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster1.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster1.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster1.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster1.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster1.ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

subcluster1.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster1.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster1.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).

Type

int

Default value

0x2

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 12

Type

int

Default value

0x1

subcluster2.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster2.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster2.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster2.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster2.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster2.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster2.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster2.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster2.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster2.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster2.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster2.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster2.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster2.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster2.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster2.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster2.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster2.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster2.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster2.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster2.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster2.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster2.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster2.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster2.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster2.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster2.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size `dcache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster2.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. `dcache-write_access_latency` must be set to 0 for per-byte latencies to be applied. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster2.etc.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x20

subcluster2.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster2.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster2.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster2.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster2.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster2.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster2.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster2.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster2.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x2

subcluster2.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster2.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster2.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster2.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster2.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster2.ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

subcluster2.ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

subcluster2.ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

subcluster2.ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

subcluster2.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster2.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster2.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster2.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster2.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster2.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster2.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).

Type

int

Default value

0x2

subcluster2.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.tlbi_stall_enabled

If true, tlbi invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster2.treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

subcluster2.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.36 ARMCortexA510CT_CortexA710CT_CortexX3CT

ARMCortexA510CT_CortexA710CT_CortexX3CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-319: IP revisions support

Revision	Quality level
CortexA510 r0p0	Full support
CortexA510 r1p3	Full support
CortexA710 r2p0	Full support
CortexX3 r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Table 3-320: Model quality changes

From	To
CortexA510 r1p3=rel	CortexA510 r0p0=rel
CortexA710 r2p0=rel	CortexA510 r1p3=rel
CortexX3 r0p0=rel	CortexA710 r2p0=rel
	CortexX3 r0p0=rel

Table 3-321: IP revision changes

From	To
CortexA510 r1p3	CortexA510 r0p0
CortexA710 r2p0	CortexA510 r1p3
CortexX3 r0p0	CortexA710 r2p0
	CortexX3 r0p0

Ports added:

- cluster_pcs_m_pchannel
- core_pcs_m_pchannel
- pmbirq

Parameters added:

- mpmmm_accumulator_multiplier
- subcluster0.tlbi_stall_enabled
- subcluster1.tlbi_stall_enabled
- subcluster2.tlbi_stall_enabled

Iris and MTI instances for ARMCortexA510CT_CortexA710CT_CortexX3CT

This model has the following Iris instances:

Table 3-322: ARMCortexA510CT_CortexA710CT_CortexX3CT Iris instances

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX3CT	Cluster_ARM_CortexA510_CortexA710_CortexX3_Heterogeneous
ARMCortexA510CT_CortexA710CT_CortexX3CT.AMU	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX3CT.AMU.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX3CT.DAP	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX3CT.DAP.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU	DSU
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_cluster	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core0	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core1	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core2	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core2.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[7]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.MMAP	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX3CT.MMAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX3CT.ext_bus	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX3CT.ext_bus.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX3CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0	Subcluster_ARM_Cortex-A510
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0	ARM_Cortex-A510
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1	Subcluster_ARM_Cortex-A710
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0	ARM_Cortex-A710
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu1.dtlb	TlbCadi

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2	Subcluster_ARM_Cortex-X3
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0	ARM_Cortex-X3
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l1licache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu2.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-323: ARMCortexA510CT_CortexA710CT_CortexX3CT MTI instances

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX3CT	ARMv8Cluster
ARMCortexA510CT_CortexA710CT_CortexX3CT.AMU	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX3CT.AMU.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX3CT.DAP	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX3CT.DAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU	DSU
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_cluster	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core0	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core1	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core2	PPUv1
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.PPU_core2.busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache	PVCache

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.shared_cache.upstream[7]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.MMAP	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX3CT.MMAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX3CT.ext_bus	PVBusLogger
ARMCortexA510CT_CortexA710CT_CortexX3CT.ext_bus.mapper	PVBusMapper
ARMCortexA510CT_CortexA710CT_CortexX3CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0	ARM_Cortex-A510
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0	ARM_Cortex-A710
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0	ARM_Cortex-X3
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l2cache	PVCache

InstanceName	ComponentName
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA710CT_CortexX3CT.subcluster2.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA510CT_CortexA710CT_CortexX3CT contains the following CADI targets:

- ARM_Cortex-A510
- ARM_Cortex-A710
- ARM_Cortex-X3
- Cluster_ARM_CortexA510_CortexA710_CortexX3_Heterogeneous
- PVMCache
- Subcluster_ARM_Cortex-A510
- Subcluster_ARM_Cortex-A710
- Subcluster_ARM_Cortex-X3
- TlbCadi

About ARMCortexA510CT_CortexA710CT_CortexX3CT

The number of cores in each subcluster is configurable using the following parameters:

subcluster0.NUM_CORES

Possible values are 1-10 (ARMCortexA510CT).

subcluster1.NUM_CORES

Possible values are 1-10 (ARMCortexA710CT).

subcluster2.NUM_CORES

Possible values are 1-10 (ARMCortexX3CT).

The total number of cores in the cluster cannot exceed 12.

Port arrays are expanded to the width needed for the maximum number of cores in each subcluster. Use the following port array indexes:

- `<port_name>[0-9]` for cores in `subcluster0`.
- `<port_name>[10-19]` for cores in `subcluster1`.
- `<port_name>[20-29]` for cores in `subcluster2`.



All instances in the Master cross trigger matrix port array, `cti[30]` must be connected, regardless of the `NUM_CORES` value used.

Core-specific parameters have the following prefixes:

- `subcluster0.cpu0` to `subcluster0.cpu9` identify cores in `subcluster0`.

- `subcluster1.cpu0` to `subcluster1.cpu9` identify cores in `subcluster1`.
- `subcluster2.cpu0` to `subcluster2.cpu9` identify cores in `subcluster2`.

For information about the cores in this model, see:

- [3.5.33 ARMCortexA510CT](#) on page 1401.
- [3.5.44 ARMCortexA710CT](#) on page 1964.
- [3.5.72 ARMCortexX3CT](#) on page 2520.

Ports for ARMCortexA510CT_CortexA710CT_CortexX3CT

Table 3-324: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AEND0MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND1MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND2MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND3MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTART0MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART1MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART2MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART3MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[30]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
<code>cluster_pcs_m_pchannel</code>	PChannel	Master	Cluster PCSM signal
<code>cluster_powerdown_out</code>	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the cluster power info

Name	Protocol	Type	Description
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[30]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[30]	Signal	Master	Timer signals to SOC
CNTHVIRQ[30]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[30]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[30]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[30]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[30]	Signal	Master	Timer signals to SOC.
commirq[30]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[30]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[30]	PChannel	Master	Core PCSM signals
core_powerdown_out[30]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[30]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[30]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[30]	Signal	Slave	Disable cryptography extensions after reset.
cti[30]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[30]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[30]	Signal	Master	No power-down request.
dbgpwrupreq[30]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[30]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components

Name	Protocol	Type	Description
gicv3_redistributor_s[30]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[30]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[30]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[30]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[30]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[30]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[30]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[30]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[30]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[30]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[30]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[30]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[30]	Signal	Slave	Virtualised FIQ.
virq[30]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[30]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA510CT_CortexA710CT_CortexX3CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAJINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x0

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

subcluster0.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 12

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster0.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster0.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster0.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster0.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster0.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster0.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster0.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster0.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster0.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster0.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster0.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster0.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster0.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster0.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.max_32bit_el

Maximum exception level supporting AArch32 modes. -1: No Support for A32 at any EL, 0: ELO supports A32. This parameter is ignored in Rev0 i.e. when revision_number = 0

Type

int

Default value

0x0

subcluster0.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

subcluster0.revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x1

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 12

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size `dcache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. `dcache-write_access_latency` must be set to 0 for per-byte latencies to be applied. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x20

subcluster1.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster1.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster1.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x1

subcluster1.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster1.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster1.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster1.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster1.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x2

subcluster1.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster1.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster1.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster1.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster1.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster1.ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

subcluster1.ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

subcluster1.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster1.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster1.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).

Type

int

Default value

0x2

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster2.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

subcluster2.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster2.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster2.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster2.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster2.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster2.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster2.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster2.cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

subcluster2.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x100000

subcluster2.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster2.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster2.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster2.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster2.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster2.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster2.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster2.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value

subcluster2.cpuX.semihosting-cwd

Base directory for semihosting file access.

Type

string

Default value**subcluster2.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster2.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster2.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster2.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster2.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster2.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster2.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster2.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster2.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster2.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster2.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.error_record_feature_register

RAS feature register values. An array of JSON objects. The JSON schema for the array is: [{"ED":0x0, "IMPDEF_3_2":0x0, "UI":0x0, "FI":0x0, "UE":0x0, "CFI":0x0, "CEC":0x0, "RP":0x0, "DUI":0x0, "CEO":0x0, "CI":0x0, "TS":0x0, "INJ":0x0, "FRX":0x0, "UC":0x0, "UEU":0x0, "UER":0x0, "UEO":0x0, "DE":0x0, "CE":0x0, "Visibility":"Core"},other_feature_register_values]. Where ED,UI,FI,CE and UE have valid values between 0x0 - 0x3. CFI and DUI have valid values 0x0, 0x2 and 0x3. CEC has valid values 0x0,0x2 or 0x4. RP,CEO,INJ,FRX,UC,UEU,UER,UEO,DE has valid values 0x0 or 0x1. CI and TS has valid values of 0x0, 0x1 and 0x2. Visibility has valid values "Core" or "Cluster"

Type

string


```

{"ED":0x2,"IMPDEF_3_2":0x0,"UI":0x2,"FI":0x2,"UE":0x1,"CFI":0x2,"CEC":0x2,"RP":0x1,"DUI":0x0,"CEC":0x0,"Visibility":"Cluster"},
{"ED":0x2,"IMPDEF_3_2":0x0,"UI":0x2,"FI":0x2,"UE":0x1,"CFI":0x2,"CEC":0x2,"RP":0x1,"DUI":0x0,"CEC":0x0,"Visibility":"Cluster"}

```

Number of claim tags

int

0x20

Maximum limit for the number of instructions implied by a Q element

int

0x1

Number of resource selector pairs

int

0x8

TRCPIDR CMOD value

int

0x0

TRCPIDR REVAND value

int

0x0

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster2.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster2.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster2.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster2.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster2.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster2.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster2.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster2.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster2.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster2.ext_abort_normal_noncacheable_read_is_sync

Synchronous reporting of normal noncacheable-read external aborts

Type

bool

Default value

0x1

subcluster2.ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x2

subcluster2.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster2.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster2.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster2.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster2.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster2.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster2.has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster2.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster2.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster2.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster2.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster2.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster2.pseudo_fault_generation_feature_register

ARMv8.4 Standard Pseudo-fault generation feature register values. JSON schema for the parameter value is: [{"OF":false, "UC":false, "UEU":false, "UER":false, "UEO":false, "DE":false, "CE":0x0, "CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":false, "NA":false}, other_pseudo-fault_generating_features_register_values]. Where OF, UC, UEU, UER, UEO, DE, CI, ER, PN, AV, MV, SYN, and R have valid false(NOT_SUPPORTED) and true(FEATURE_CONTROLLABLE), where CE can have 0(NOT_SUPPORTED), 1(NONSPECIFIC_CE_SUPPORTED) and 3(TRANSIENT_OR_PERSISTENT_CE_SUPPORTED) and NA can have false(component fakes detection on next access) or true(component fakes detection spontaneously). Effective only when ERXFR's INJ field allows it or has_ras_fault_injection is true.

Type

string

Default value

```
[{"OF":true, "UC":true, "UEU":false, "UER":false, "UEO":false, "DE":0x1,
"CE":0x1,"CI":true, "ER":false, "PN":true, "AV":false, "MV":true, "SYN":true,
```

```
"R":true},{ "OF":false, "UC":true, "UEU":false, "UER":false, "UEO":false, "DE":0x1,
"CE":0x1,"CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":true}]
```

subcluster2.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster2.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster2.trace_physical_registers_when_host_virtualisation_enabled

When host virtualisation is enabled, trace sysreg accesses to physical register accessed (0=disabled, 1=Trace only ELR/SPSR_EL1 as ELR/SPSR_EL2, 2=Trace all redirected registers as physical registers)

Type

int

Default value

0x1

subcluster2.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster2.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.37 ARMCortexA510CT_CortexA715CT_CortexX3CT

ARMCortexA510CT_CortexA715CT_CortexX3CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-325: IP revisions support

Revision	Quality level
CortexA510 r0p0	Full support
CortexA510 r1p3	Full support
CortexA715 r1p2	Full support
CortexX3 r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Table 3-326: Model quality changes

From	To
CortexA510 r1p3=rel	CortexA510 r0p0=rel
CortexA715 r1p2=rel	CortexA510 r1p3=rel
CortexX3 r0p0=rel	CortexA715 r1p2=rel
	CortexX3 r0p0=rel

Table 3-327: IP revision changes

From	To
CortexA510 r1p3	CortexA510 r0p0
CortexA715 r1p2	CortexA510 r1p3
CortexX3 r0p0	CortexA715 r1p2
	CortexX3 r0p0

Ports added:

- cluster_pcs_m_pchannel
- core_pcs_m_pchannel

Parameters added:

- mpmmm_accumulator_multiplier
- subcluster0.tlbi_stall_enabled
- subcluster1.tlbi_stall_enabled
- subcluster2.tlbi_stall_enabled

Iris and MTI instances for ARMCortexA510CT_CortexA715CT_CortexX3CT

This model has the following Iris instances:

Table 3-328: ARMCortexA510CT_CortexA715CT_CortexX3CT Iris instances

InstanceName	ComponentName
ARMCortexA510CT_CortexA715CT_CortexX3CT	Cluster_ARM_CortexA510_CortexA715_CortexX3_Heterogeneous
ARMCortexA510CT_CortexA715CT_CortexX3CT.AMU	PVBusLogger
ARMCortexA510CT_CortexA715CT_CortexX3CT.AMU.mapper	PVBusMapper
ARMCortexA510CT_CortexA715CT_CortexX3CT.DAP	PVBusLogger
ARMCortexA510CT_CortexA715CT_CortexX3CT.DAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU	DSU
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_cluster	PPUv1
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core0	PPUv1
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core1	PPUv1

InstanceName	ComponentName
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core2	PPUv1
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core2.busslave	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[7]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.MMAP	PVBusLogger
ARMCortexA510CT_CortexA715CT_CortexX3CT.MMAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA715CT_CortexX3CT.ext_bus	PVBusLogger
ARMCortexA510CT_CortexA715CT_CortexX3CT.ext_bus.mapper	PVBusMapper
ARMCortexA510CT_CortexA715CT_CortexX3CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0	Subcluster_ARM_Cortex-A510
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0	ARM_Cortex-A510
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.dtlb	TlbCadi

InstanceName	ComponentName
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1	Subcluster_ARM_Cortex-A715
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0	ARM_Cortex-A715
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l1licache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu1.dtlb	TlbCadi
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2	Subcluster_ARM_Cortex-X3
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0	ARM_Cortex-X3
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l1dcache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu2.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-329: ARMCortexA510CT_CortexA715CT_CortexX3CT MTI instances

InstanceName	ComponentName
ARMCortexA510CT_CortexA715CT_CortexX3CT	ARMv8Cluster
ARMCortexA510CT_CortexA715CT_CortexX3CT.AMU	PVBusLogger
ARMCortexA510CT_CortexA715CT_CortexX3CT.AMU.mapper	PVBusMapper
ARMCortexA510CT_CortexA715CT_CortexX3CT.DAP	PVBusLogger
ARMCortexA510CT_CortexA715CT_CortexX3CT.DAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU	DSU
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_cluster	PPUv1
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core0	PPUv1
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core1	PPUv1
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core2	PPUv1
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.PPU_core2.busslave	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.shared_cache.upstream[7]	PVBusSlave

InstanceName	ComponentName
ARMCortexA510CT_CortexA715CT_CortexX3CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.MMAP	PVBusLogger
ARMCortexA510CT_CortexA715CT_CortexX3CT.MMAP.mapper	PVBusMapper
ARMCortexA510CT_CortexA715CT_CortexX3CT.ext_bus	PVBusLogger
ARMCortexA510CT_CortexA715CT_CortexX3CT.ext_bus.mapper	PVBusMapper
ARMCortexA510CT_CortexA715CT_CortexX3CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0	ARM_Cortex-A510
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0	ARM_Cortex-A715
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0	ARM_Cortex-X3
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.UTLB	TLB
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l1dcache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l1icache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l2cache	PVCache
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA510CT_CortexA715CT_CortexX3CT.subcluster2.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA510CT_CortexA715CT_CortexX3CT contains the following CADI targets:

- ARM_Cortex-A510
- ARM_Cortex-A715
- ARM_Cortex-X3
- Cluster_ARM_CortexA510_CortexA715_CortexX3_Heterogeneous

- PVCache
- Subcluster_ARM_Cortex-A510
- Subcluster_ARM_Cortex-A715
- Subcluster_ARM_Cortex-X3
- TlbCadi

About ARMCortexA510CT_CortexA715CT_CortexX3CT

The number of cores in each subcluster is configurable using the following parameters:

subcluster0.NUM_CORES

Possible values are 1-10 (ARMCortexA510CT).

subcluster1.NUM_CORES

Possible values are 1-10 (ARMCortexA715CT).

subcluster2.NUM_CORES

Possible values are 1-10 (ARMCortexX3CT).

The total number of cores in the cluster cannot exceed 12.

Port arrays are expanded to the width needed for the maximum number of cores in each subcluster. Use the following port array indexes:

- `<port_name>[0-9]` for cores in `subcluster0`.
- `<port_name>[10-19]` for cores in `subcluster1`.
- `<port_name>[20-29]` for cores in `subcluster2`.



All instances in the Master cross trigger matrix port array, `cti[30]` must be connected, regardless of the `NUM_CORES` value used.

Core-specific parameters have the following prefixes:

- `subcluster0.cpu0` to `subcluster0.cpu9` identify cores in `subcluster0`.
- `subcluster1.cpu0` to `subcluster1.cpu9` identify cores in `subcluster1`.
- `subcluster2.cpu0` to `subcluster2.cpu9` identify cores in `subcluster2`.

For information about the cores in this model, see:

- [3.5.33 ARMCortexA510CT](#) on page 1401.
- [3.5.45 ARMCortexA715CT](#) on page 1992.
- [3.5.72 ARMCortexX3CT](#) on page 2520.

Ports for ARMCortexA510CT_CortexA715CT_CortexX3CT

Table 3-330: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[30]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcsn_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[30]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[30]	Signal	Master	Timer signals to SOC
CNTHVIRQ[30]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[30]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[30]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[30]	Signal	Master	Timer signals to SOC.

Name	Protocol	Type	Description
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[30]	Signal	Master	Timer signals to SOC.
commirq[30]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[30]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[30]	PChannel	Master	Core PCSM signals
core_powerdown_out[30]	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the core power info
coreerrirq[30]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[30]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[30]	Signal	Slave	Disable cryptography extensions after reset.
cti[30]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[30]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[30]	Signal	Master	No power-down request.
dbgpwrupreq[30]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[30]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPUs that can be used to reset GICCLK domain components
gicv3_redistributor_s[30]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[30]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[30]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[30]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.

Name	Protocol	Type	Description
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[30]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[30]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[30]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[30]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[30]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[30]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[30]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[30]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[30]	Signal	Slave	Virtualised FIQ.
virq[30]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[30]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA510CT_CortexA715CT_CortexX3CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

subcluster0.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 12

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster0.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster0.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster0.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster0.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster0.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster0.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster0.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster0.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster0.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster0.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster0.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster0.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster0.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster0.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.max_32bit_el

Maximum exception level supporting AArch32 modes. -1: No Support for A32 at any EL, 0: ELO supports A32. This parameter is ignored in Rev0 i.e. when revision_number = 0

Type

int

Default value

0x0

subcluster0.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

subcluster0.revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x1

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.error_record_feature_register

RAS feature register values. An array of JSON objects. The JSON schema for the array is:
[{"ED":0x0, "IMPDEF_3_2":0x0, "UI":0x0, "FI":0x0, "UE":0x0, "CFI":0x0, "CEC":0x0, "RP":0x0, "DUI":0x0, "CEO":0x0, "CI":0x0, "TS":0x0, "INJ":0x0, "FRX":0x0, "UC":0x0, "UEU":0x0, "UER":0x0, "UEO":0x0, "DE":0x0, "CE":0x0, "Visibility":"Core"},other_feature_register_values].
Where ED,UI,FI,CE and UE have valid values between 0x0 - 0x3. CFI and
DUI have valid values 0x0, 0x2 and 0x3. CEC has valid values 0x0,0x2 or 0x4.
RP,CEO,INJ,FRX,UC,UEU,UER,UEO,DE has valid values 0x0 or 0x1. CI and TS has valid values
of 0x0, 0x1 and 0x2. Visibility has valid values "Core" or "Cluster"

Type

string

Default value

[{"ED":0x2,"IMPDEF_3_2":0x1,"UI":0x2,"FI":0x2,"UE":0x1,"CFI":0x2,"CEC":0x2,"RP":0x1,"DUI":0x0,"CE":0x0,"Visibility":"Cluster"},
{"ED":0x2,"UI":0x2,"FI":0x2,"UE":0x1,"CFI":0x2,"CEC":0x2,"RP":0x1,"DUI":0x0,"CEO":0x0,"INJ":0x1,"CI":0x0,"TS":0x0,"FRX":0x0,"UC":0x0,"UEU":0x0,"UER":0x0,"UEO":0x0,"DE":0x0,"CE":0x0}],other_feature_register_values]

subcluster1.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster1.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster1.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster1.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster1.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster1.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster1.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster1.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x1

subcluster1.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster1.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster1.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster1.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster1.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster1.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster1.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster1.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster1.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster1.has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster1.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster1.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster1.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster1.pseudo_fault_generation_feature_register

ARMv8.4 Standard Pseudo-fault generation feature register values. JSON schema for the parameter value is: [{"OF":false, "UC":false, "UEU":false, "UER":false, "UEO":false, "DE":false, "CE":0x0, "CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":false, "NA":false}, other_pseudo-fault_generating_features_register_values]. Where OF, UC, UEU, UER, UEO, DE, CI, ER, PN, AV, MV, SYN, and R have valid false(NOT_SUPPORTED) and true(FEATURE_CONTROLLABLE), where CE can have 0(NOT_SUPPORTED), 1(NONSPECIFIC_CE_SUPPORTED) and 3(TRANSIENT_OR_PERSISTENT_CE_SUPPORTED) and NA can have false(component fakes detection on next access) or true(component fakes detection spontaneously). Effective only when ERXFR's INJ field allows it or has_ras_fault_injection is true.

Type

string

Default value

```
[{"OF":true, "UC":true, "UEU":false, "UER":false, "UEO":false, "DE":0x1,
"CE":0x1,"CI":true, "ER":false, "PN":true, "AV":false, "MV":true, "SYN":true,
"R":true},{ "OF":false, "UC":true, "UEU":false, "UER":false, "UEO":false, "DE":0x1,
"CE":0x1,"CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":true}]
```

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster2.NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

subcluster2.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster2.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster2.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster2.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster2.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster2.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster2.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster2.cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

subcluster2.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x100000

subcluster2.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster2.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster2.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster2.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster2.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster2.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster2.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster2.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster2.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster2.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster2.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster2.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster2.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster2.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster2.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster2.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster2.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster2.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster2.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster2.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.error_record_feature_register

RAS feature register values. An array of JSON objects. The JSON schema for the array is:

```
[{"ED":0x0, "IMPDEF_3_2":0x0, "UI":0x0, "FI":0x0, "UE":0x0, "CFI":0x0, "CEC":0x0, "RP":0x0, "DUI":0x0, "CEO":0x0, "CI":0x0, "TS":0x0, "INJ":0x0, "FRX":0x0, "UC":0x0, "UEU":0x0, "UER":0x0, "UEO":0x0, "DE":0x0, "CE":0x0, "Visibility":"Core"},other_feature_register_values].
```

Where ED,UI,FI,CE and UE have valid values between 0x0 - 0x3. CFI and

DUI have valid values 0x0, 0x2 and 0x3. CEC has valid values 0x0,0x2 or 0x4.

RP,CEO,INJ,FRX,UC,UEU,UER,UEO,DE has valid values 0x0 or 0x1. CI and TS has valid values of 0x0, 0x1 and 0x2. Visibility has valid values "Core" or "Cluster"

Type

string

Default value

```
[{"ED":0x2,"IMPDEF_3_2":0x0,"UI":0x2,"FI":0x2,"UE":0x1,"CFI":0x2,"CEC":0x2,"RP":0x1,"DUI":0x0,"CE":0x0,"Visibility":"Cluster"},
{"ED":0x2,"IMPDEF_3_2":0x0,"UI":0x2,"FI":0x2,"UE":0x1,"CFI":0x2,"CEC":0x2,"RP":0x1,"DUI":0x0,"CE":0x0,"Visibility":"Cluster"}]
```

subcluster2.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x20

subcluster2.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster2.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster2.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster2.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster2.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster2.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster2.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster2.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster2.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster2.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster2.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster2.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster2.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster2.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster2.ext_abort_normal_noncacheable_read_is_sync

Synchronous reporting of normal noncacheable-read external aborts

Type

bool

Default value

0x1

subcluster2.ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x2

subcluster2.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster2.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster2.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster2.has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

subcluster2.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster2.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster2.has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster2.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster2.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

subcluster2.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster2.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster2.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster2.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster2.pseudo_fault_generation_feature_register

ARMv8.4 Standard Pseudo-fault generation feature register values. JSON schema for the parameter value is: [{"OF":false, "UC":false, "UEU":false, "UER":false, "UEO":false, "DE":false, "CE":0x0, "CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":false, "NA":false}, other_pseudo-fault_generating_features_register_values]. Where OF, UC,

UEU, UER, UEO, DE, CI, ER, PN, AV, MV, SYN, and R have valid false(NOT_SUPPORTED) and true(FEATURE_CONTROLLABLE), where CE can have 0(NOT_SUPPORTED), 1(NONSPECIFIC_CE_SUPPORTED) and 3(TRANSIENT_OR_PERSISTENT_CE_SUPPORTED) and NA can have false(component fakes detection on next access) or true(component fakes detection spontaneously). Effective only when ERXFR's INJ field allows it or has_ras_fault_injection is true.

Type

string

Default value

```
[{"OF":true, "UC":true, "UEU":false, "UER":false, "UEO":false, "DE":0x1,
"CE":0x1,"CI":true, "ER":false, "PN":true, "AV":false, "MV":true, "SYN":true,
"R":true},{ "OF":false, "UC":true, "UEU":false, "UER":false, "UEO":false, "DE":0x1,
"CE":0x1,"CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":true}]
```

subcluster2.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster2.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster2.trace_physical_registers_when_host_virtualisation_enabled

When host virtualisation is enabled, trace sysreg accesses to physical register accessed (0=disabled, 1=Trace only ELR/SPSR_EL1 as ELR/SPSR_EL2, 2=Trace all redirected registers as physical registers)

Type

int

Default value

0x1

subcluster2.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster2.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.38 **ARMCortexA520AECT**

ARMCortexA520AECT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-331: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Ports added:

- cluster_pcs_m_pchannel
- core_pcs_m_pchannel

Parameters added:

- core_complex_mapping
- mpmu_accumulator_multiplier
- tlbi_stall_enabled

Iris and MTI instances for ARM Cortex-A520AECT

This model has the following Iris instances:

Table 3-332: ARM Cortex-A520AECT Iris instances

InstanceName	ComponentName
ARM Cortex-A520AECT	Cluster_ARM_Cortex-A520AE
ARM Cortex-A520AECT.AMU	PVBusLogger
ARM Cortex-A520AECT.AMU.mapper	PVBusMapper
ARM Cortex-A520AECT.DAP	PVBusLogger
ARM Cortex-A520AECT.DAP.mapper	PVBusMapper
ARM Cortex-A520AECT.DSU	DSU
ARM Cortex-A520AECT.DSU.PPU_cluster	PPUv1
ARM Cortex-A520AECT.DSU.PPU_cluster.busslave	PVBusSlave
ARM Cortex-A520AECT.DSU.PPU_core0	PPUv1
ARM Cortex-A520AECT.DSU.PPU_core0.busslave	PVBusSlave
ARM Cortex-A520AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM Cortex-A520AECT.DSU.mpam_busslave	PVBusSlave
ARM Cortex-A520AECT.DSU.shared_cache	PVCache
ARM Cortex-A520AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM Cortex-A520AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARM Cortex-A520AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARM Cortex-A520AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARM Cortex-A520AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARM Cortex-A520AECT.DSU.utility_slave[0]	PVBusSlave
ARM Cortex-A520AECT.MMAP	PVBusLogger
ARM Cortex-A520AECT.MMAP.mapper	PVBusMapper
ARM Cortex-A520AECT.cpu0	ARM_Cortex-A520AE
ARM Cortex-A520AECT.cpu0.UTLB	TLB
ARM Cortex-A520AECT.cpu0.dtlb	TlbCadi
ARM Cortex-A520AECT.cpu0.l1dcache	PVCache
ARM Cortex-A520AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM Cortex-A520AECT.cpu0.l1icache	PVCache
ARM Cortex-A520AECT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM Cortex-A520AECT.cpu0.l2cache	PVCache
ARM Cortex-A520AECT.cpu0.l2cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA520AECT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520AECT.ext_bus	PVBusLogger
ARMCortexA520AECT.ext_bus.mapper	PVBusMapper
ARMCortexA520AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-333: ARMCortexA520AECT MTI instances

InstanceName	ComponentName
ARMCortexA520AECT	ARMv8Cluster
ARMCortexA520AECT.AMU	PVBusLogger
ARMCortexA520AECT.AMU.mapper	PVBusMapper
ARMCortexA520AECT.DAP	PVBusLogger
ARMCortexA520AECT.DAP.mapper	PVBusMapper
ARMCortexA520AECT.DSU	DSU
ARMCortexA520AECT.DSU.PPU_cluster	PPUv1
ARMCortexA520AECT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA520AECT.DSU.PPU_core0	PPUv1
ARMCortexA520AECT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA520AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA520AECT.DSU.mpam_busslave	PVBusSlave
ARMCortexA520AECT.DSU.shared_cache	PVCache
ARMCortexA520AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA520AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA520AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520AECT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520AECT.MMAP	PVBusLogger
ARMCortexA520AECT.MMAP.mapper	PVBusMapper
ARMCortexA520AECT.cpu0	ARM_Cortex-A520AE
ARMCortexA520AECT.cpu0.UTLB	TLB
ARMCortexA520AECT.cpu0.l1dcache	PVCache
ARMCortexA520AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520AECT.cpu0.l1licache	PVCache
ARMCortexA520AECT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA520AECT.cpu0.l2cache	PVCache
ARMCortexA520AECT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520AECT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520AECT.ext_bus	PVBusLogger

InstanceName	ComponentName
ARMCortexA520AECT.ext_bus.mapper	PVBusMapper
ARMCortexA520AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA520AECT contains the following CADI targets:

- ARM_Cortex-A520AE
- Cluster_ARM_Cortex-A520AE
- PVCache
- TlbCadi

AE-specific features implemented

The only AE-specific features implemented in the model are Split-Lock and Lock-Step, with the following limitations:

- The implementation of Split-Lock and Lock-Step in the model is simply a halving of the number of cores available.
- Lock-step is implemented by a parameter that can be true or false for the entire execution only, rather than by a signal that can be enabled during execution.

As the model does not implement true Split-Lock functionality, the ports and parameters are contiguous. For example, for a 4-core CPU with Split-Lock enabled, `cpu0` and `cpu1` identify the available cores and associated ports, not `cpu0` and `cpu2`.

- Hybrid mode is not modeled in the DSU.

Ports for ARMCortexA520AECT

Table 3-334: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.

Name	Protocol	Type	Description
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[14]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[14]	Signal	Master	Timer signals to SOC
CNTHVIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHVSIQ[14]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[14]	Signal	Master	Timer signals to SOC.
CNTPSIQ[14]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[14]	Signal	Master	Timer signals to SOC.
commirq[14]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[14]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[14]	PChannel	Master	Core PCSM signals
core_powerdown_out[14]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[14]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[14]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[14]	Signal	Slave	Disable cryptography extensions after reset.
cti[14]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[14]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.

Name	Protocol	Type	Description
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[14]	Signal	Master	No power-down request.
dbgpwrupreq[14]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[14]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[14]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[14]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmuirq[14]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[14]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[14]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[14]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[14]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[14]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.

Name	Protocol	Type	Description
ticks[14]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[14]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[14]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[14]	Signal	Slave	Virtualised FIQ.
virq[14]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[14]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA520AECT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance

of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

cluster_split_lock_config

Default SPLIT/LOCKED config. The valid values are: 1 - Only LOCKED mode support, 4 - Only SPLIT mode support, 5 - SPLIT or MIXED mode support. Valid only when enable_ae_features is true.

Type

int

Default value

0x1

core_cache_protection

core_cache_protection can change ERROFR, ERROPFGF and ERROPFGCTL fields. Possible values are: -1:Not implemented (by default), 0:Disabled, 1:Enabled

Type

int

Default value

0x1

core_complex_mapping

Defines Complex descriptions for platforms that support several Cores per Complex like Cortex-A510. JSON format: {"complex0": { "cores" : [0, 1], "l2-cache" : {"exists":1, "size":16MB}}, ... , "complexN": { "cores" : [<core_list>], "l2-cache" : {"exists":1, "size":16MB}}} where <core_list> is the list of cores in the complexN. Effective only when the parameter value is not empty.

Type

string

Default value

```
{"complex0": { "cores": [0, 1], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex1":
{ "cores": [2, 3], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex2": { "cores": [4, 5], "l2-
cache" : {"exists":1, "size":"16MB"}}, "complex3": { "cores": [6, 7], "l2-cache" : {"exists":1,
"size":"16MB"}}, "complex4": { "cores": [8, 9], "l2-cache" : {"exists":1, "size":"16MB"}},
```

```
"complex5": { "cores": [10, 11], "l2-cache" :{"exists":1, "size":"16MB"}}, "complex6":
{ "cores": [12, 13], "l2-cache" :{"exists":1, "size":"16MB"}} }
```

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_lock_step

Whether the core is configured in Dual Core Lock Step mode (FEAT_DCLS)

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCSRSTA_FORCED_EXCEP**

TRCSRSTA value for a forcibly traced exception

Type

bool

Default value

0x0

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_actlr2

If true ACLTR2 exists and ACTLR2(NS) is aliased to ACTLR_EL1[63:32]

Type

bool

Default value

0x1

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_impdef_transient_fault_protection

Support the Transient Fault Protection (TFP) flop parity errors through RAS registers (FEAT_TFP)

Type

bool

Default value

0x1

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_bus_width_in_bytes

L3 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_bus_width_in_bytes

L3 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use $(n * \text{accumulator value})$ to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_acp

Number of ACP ports

Type

int

Default value

0x0

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

ras_extra_configurations

Miscellaneous configurations for error records. An array of JSON objects. Note for ERXCTLR_EL1 register it only allows to define the mask value for the IMPDEF fields,

ie bits [63:32] and bit 1, but its reset value applies on all fields. Note for ERXMISCN masks - these are 64 bit masks covering the 64 bit registers ERXMISCN_EL1. E.g.
 [{"Index": 0, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXMISC1_mask": 0x0, "ERXMISC1_reset": 0x0, "ERXMISC2_mask": 0x0, "ERXMISC2_reset": 0x0, "ERXMISC3_mask": 0x0, "ERXMISC3_reset": 0x0, "ERXCTLR_EL1_mask": 0x0, "ERXCTLR_EL1_reset": 0x0}, {"Index": 1, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXSTATUS_IERR_mask": 0x300}]

Type

string

Default value

```
[ { "Index": 1, "ERXMISCO_mask": 0xFFFFc0003fc3, "ERXMISC1_mask": 0x03F870003FF30f07, "ERXPFGCTL_reset": 0x1000 }, { "Index": 2, "ERXMISCO_mask": 0xFFFFe007ffc0, "ERXMISCO_reset": 0x2, "ERXSTATUS_IERR_mask": 0x300 , "ERXMISC1_mask": 0x0FF8700fFFF31f0f, "ERXPFGCTL_reset": 0x1000 } ]
```

ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.39 **ARMCortexA520CT**

ARMCortexA520CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-335: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Ports added:

- cluster_pcs_m_pchannel
- core_pcs_m_pchannel

Parameters added:

- core_complex_mapping
- mpm_accumulator_multiplier
- tlbi_stall_enabled

Iris and MTI instances for ARMCortexA520CT

This model has the following Iris instances:

Table 3-336: ARMCortexA520CT Iris instances

InstanceName	ComponentName
ARMCortexA520CT	Cluster_ARM_Cortex-A520
ARMCortexA520CT.AMU	PVBusLogger
ARMCortexA520CT.AMU.mapper	PVBusMapper
ARMCortexA520CT.DAP	PVBusLogger
ARMCortexA520CT.DAP.mapper	PVBusMapper
ARMCortexA520CT.DSU	DSU
ARMCortexA520CT.DSU.PPU_cluster	PPUv1
ARMCortexA520CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA520CT.DSU.PPU_core0	PPUv1
ARMCortexA520CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA520CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA520CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA520CT.DSU.shared_cache	PVCache
ARMCortexA520CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA520CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA520CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520CT.MMAP	PVBusLogger
ARMCortexA520CT.MMAP.mapper	PVBusMapper
ARMCortexA520CT.cpu0	ARM_Cortex-A520
ARMCortexA520CT.cpu0.UTLB	TLB
ARMCortexA520CT.cpu0.dtlb	TlbCadi
ARMCortexA520CT.cpu0.l1dcache	PVCache

InstanceName	ComponentName
ARMCortexA520CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT.cpu0.l1icache	PVCache
ARMCortexA520CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT.cpu0.l2cache	PVCache
ARMCortexA520CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT.ext_bus	PVBusLogger
ARMCortexA520CT.ext_bus.mapper	PVBusMapper
ARMCortexA520CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-337: ARMCortexA520CT MTI instances

InstanceName	ComponentName
ARMCortexA520CT	ARMv8Cluster
ARMCortexA520CT.AMU	PVBusLogger
ARMCortexA520CT.AMU.mapper	PVBusMapper
ARMCortexA520CT.DAP	PVBusLogger
ARMCortexA520CT.DAP.mapper	PVBusMapper
ARMCortexA520CT.DSU	DSU
ARMCortexA520CT.DSU.PPU_cluster	PPUv1
ARMCortexA520CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA520CT.DSU.PPU_core0	PPUv1
ARMCortexA520CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA520CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA520CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA520CT.DSU.shared_cache	PVCache
ARMCortexA520CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA520CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA520CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520CT.MMAP	PVBusLogger
ARMCortexA520CT.MMAP.mapper	PVBusMapper
ARMCortexA520CT.cpu0	ARM_Cortex-A520
ARMCortexA520CT.cpu0.UTLB	TLB
ARMCortexA520CT.cpu0.l1dcache	PVCache
ARMCortexA520CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT.cpu0.l1icache	PVCache

InstanceName	ComponentName
ARMCortexA520CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT.cpu0.l2cache	PVCache
ARMCortexA520CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT.ext_bus	PVBusLogger
ARMCortexA520CT.ext_bus.mapper	PVBusMapper
ARMCortexA520CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA520CT contains the following CADI targets:

- ARM_Cortex-A520
- Cluster_ARM_Cortex-A520
- PVCache
- TlbCadi

About ARMCortexA520CT

The model supports the following features:

- L2 cache is supported at the per-core level only and there is no implementation of Core-Complex with shared L2 cache yet.
- Internal PPU support is present.
- A P-Channel for the cluster and for each core.
- BROADCASTPERSIST pin.
- Optional peripheral port.
- Memory-mapped register access to MPAM.
- Per-core clock.
- Utility bus.

Support for the following features is planned for a future release:

- DSU-120 system features are not fully implemented.
- Transporting architectural metadata tags over TLM2 (OSCI) and AMBA-PV (TLM2 extensions from Arm) bus protocols.
- Core-Complex.
- BROADCASTCACHEMAINTPOU pin
- COREINSTRRET andCOREINSTRRUN signals.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not supported but signals are implemented.

The following features will not be implemented:

- DynamiQ™ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Cache stashing capability.

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA520CT

Table 3-338: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[14]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[14]	Signal	Master	Timer signals to SOC
CNTHVIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[14]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[14]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[14]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[14]	Signal	Master	Timer signals to SOC.
commirq[14]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[14]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[14]	PChannel	Master	Core PCSM signals
core_powerdown_out[14]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[14]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[14]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[14]	Signal	Slave	Disable cryptography extensions after reset.
cti[14]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[14]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[14]	Signal	Master	No power-down request.
dbgprupreq[14]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.

Name	Protocol	Type	Description
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[14]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[14]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[14]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmuirq[14]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[14]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[14]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[14]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[14]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[14]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[14]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[14]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[14]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[14]	Signal	Slave	Virtualised FIQ.
virq[14]	Signal	Slave	Virtualised IRQ.

Name	Protocol	Type	Description
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[14]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A520CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAIN

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_cache_protection

core_cache_protection can change ERROFR, ERROPFGF and ERROPFGCTL fields. Possible values are: -1:Not implemented (by default), 0:Disabled, 1:Enabled

Type

int

Default value

0x1

core_complex_mapping

Defines Complex descriptions for platforms that support several Cores per Complex like Cortex-A510. JSON format: {"complex0": { "cores" : [0, 1], "l2-cache" : {"exists":1, "size":16MB}}, ... , "complexN": { "cores" : [<core_list>], "l2-cache" : {"exists":1, "size":16MB}}} where <core_list> is the list of cores in the complexN. Effective only when the parameter value is not empty.

Type

string

Default value

```
{"complex0": { "cores": [0, 1], "l2-cache" :{"exists":1, "size":"16MB"}}, "complex1":
{ "cores": [2, 3], "l2-cache" :{"exists":1, "size":"16MB"}}, "complex2": { "cores": [4, 5], "l2-
cache" :{"exists":1, "size":"16MB"}}, "complex3": { "cores": [6, 7], "l2-cache" :{"exists":1,
"size":"16MB"}}, "complex4": { "cores": [8, 9], "l2-cache" :{"exists":1, "size":"16MB"}},
"complex5": { "cores": [10, 11], "l2-cache" :{"exists":1, "size":"16MB"}}, "complex6":
{ "cores": [12, 13], "l2-cache" :{"exists":1, "size":"16MB"}} }
```

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_actlr2

If true ACLTR2 exists and ACTLR2(NS) is aliased to ACTLR_EL1[63:32]

Type

bool

Default value

0x1

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_bus_width_in_bytes

L3 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_bus_width_in_bytes

L3 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_acp

Number of ACP ports

Type

int

Default value

0x0

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

ras_extra_configurations

Miscellaneous configurations for error records. An array of JSON objects. Note for ERXCTLR_EL1 register it only allows to define the mask value for the IMPDEF fields, ie bits [63:32] and bit 1, but its reset value applies on all fields. Note for ERXMISCN masks - these are 64 bit masks covering the 64 bit registers ERXMISCN_EL1. E.g. [{"Index": 0, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXMISC1_mask": 0x0, "ERXMISC1_reset": 0x0, "ERXMISC2_mask": 0x0, "ERXMISC2_reset": 0x0, "ERXMISC3_mask": 0x0, "ERXMISC3_reset": 0x0, "ERXCTLR_EL1_mask": 0x0, "ERXCTLR_EL1_reset": 0x0}, {"Index": 1, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXSTATUS_IERR_mask": 0x300}]

Type

string

Default value

[{ "Index": 1, "ERXMISCO_mask": 0xFFFFc0003fc3, "ERXMISC1_mask": 0x03F870003FF30f07, "ERXPFGCTL_reset": 0x1000 }, { "Index": 2, "ERXMISCO_mask": 0xFFFFe007ffc0, "ERXMISCO_reset": 0x2, "ERXSTATUS_IERR_mask": 0x300 , "ERXMISC1_mask": 0x0FF8700fFFF31f0f, "ERXPFGCTL_reset": 0x1000 }]

ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.40 ARMCortexA520CT_CortexA720CT

ARMCortexA520CT_CortexA720CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-339: IP revisions support

Revision	Quality level
CortexA520 r0p1	Full support
CortexA720 r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Table 3-340: Model quality changes

From	To
CortexA520 r0p0=rel	CortexA520 r0p1=rel
CortexA720 r0p0=rel	CortexA720 r0p1=rel

Table 3-341: IP revision changes

From	To
CortexA520 r0p1	CortexA520 r0p1
CortexA720 r0p1	CortexA720 r0p1

Ports added:

- cluster_pcs_m_pchannel
- core_pcs_m_pchannel

Parameters added:

- core_complex_mapping
- mpm_accumulator_multiplier
- subcluster0.tlbi_stall_enabled
- subcluster1.tlbi_stall_enabled

Iris and MTI instances for ARMCortexA520CT_CortexA720CT

This model has the following Iris instances:

Table 3-342: ARMCortexA520CT_CortexA720CT Iris instances

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT	Cluster_ARM_CortexA520_CortexA720_Heterogeneous
ARMCortexA520CT_CortexA720CT.AMU	PVBusLogger

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT.AMU.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT.DAP	PVBusLogger
ARMCortexA520CT_CortexA720CT.DAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT.DSU	DSU
ARMCortexA520CT_CortexA720CT.DSU.PPU_cluster	PPUv1
ARMCortexA520CT_CortexA720CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.PPU_core0	PPUv1
ARMCortexA520CT_CortexA720CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.PPU_core1	PPUv1
ARMCortexA520CT_CortexA720CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA520CT_CortexA720CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache	PVCache
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.MMAP	PVBusLogger
ARMCortexA520CT_CortexA720CT.MMAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT.ext_bus	PVBusLogger
ARMCortexA520CT_CortexA720CT.ext_bus.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA520CT_CortexA720CT.subcluster0	Subcluster_ARM_Cortex-A520
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0	ARM_Cortex-A520
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster1	Subcluster_ARM_Cortex-A720
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0	ARM_Cortex-A720
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster1.cpu1.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-343: ARMCortexA520CT_CortexA720CT MTI instances

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT	ARMv8Cluster
ARMCortexA520CT_CortexA720CT.AMU	PVBusLogger
ARMCortexA520CT_CortexA720CT.AMU.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT.DAP	PVBusLogger
ARMCortexA520CT_CortexA720CT.DAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT.DSU	DSU
ARMCortexA520CT_CortexA720CT.DSU.PPU_cluster	PPUv1
ARMCortexA520CT_CortexA720CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.PPU_core0	PPUv1
ARMCortexA520CT_CortexA720CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.PPU_core1	PPUv1
ARMCortexA520CT_CortexA720CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA520CT_CortexA720CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache	PVCache
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[3]	PVBusSlave

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA520CT_CortexA720CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.MMAP	PVBusLogger
ARMCortexA520CT_CortexA720CT.MMAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT.ext_bus	PVBusLogger
ARMCortexA520CT_CortexA720CT.ext_bus.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0	ARM_Cortex-A520
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0	ARM_Cortex-A720
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l1licache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA520CT_CortexA720CT contains the following CADI targets:

- ARM_Cortex-A520
- ARM_Cortex-A720
- Cluster_ARM_CortexA520_CortexA720_Heterogeneous
- PVCache
- Subcluster_ARM_Cortex-A520
- Subcluster_ARM_Cortex-A720
- TlbCadi

About ARMCortexA520CT_CortexA720CT

The number of cores in each subcluster is configurable using the following parameters:

subcluster0.NUM_CORES

Possible values are 1-13 (ARMCortexA520CT).

subcluster1.NUM_CORES

Possible values are 1-13 (ARMCortexA720CT).

The total number of cores in the cluster cannot exceed 14.

Port arrays are expanded to the width needed for the maximum number of cores in each subcluster. Use the following port array indexes:

- `<port_name>[0-12]` for cores in `subcluster0`.
- `<port_name>[13-25]` for cores in `subcluster1`.



All instances in the Master cross trigger matrix port array, `cti[26]` must be connected, regardless of the `NUM_CORES` value used.

Core-specific parameters have the following prefixes:

- `subcluster0.cpu0` to `subcluster0.cpu12` identify cores in `subcluster0`.
- `subcluster1.cpu0` to `subcluster1.cpu12` identify cores in `subcluster1`.

For information about the cores in this model, see:

- [3.5.39 ARMCortexA520CT](#) on page 1703.
- [3.5.47 ARMCortexA720CT](#) on page 2054.

Ports for ARMCortexA520CT_CortexA720CT

Table 3-344: Ports

Name	Protocol	Type	Description
<code>acp_s[2]</code>	PVBus	Slave	AXI ACP slave port.
<code>AEND0MP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND1MP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND2MP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND3MP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTART0MP</code>	<code>Value_64</code>	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART1MP</code>	<code>Value_64</code>	Slave	Port to obtain start address of valid peripheral address range (inclusive).

Name	Protocol	Type	Description
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[26]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcsmp_channel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[26]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[26]	Signal	Master	Timer signals to SOC
CNTHVIRQ[26]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[26]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[26]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[26]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[26]	Signal	Master	Timer signals to SOC.
commirq[26]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[26]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmp_channel[26]	PChannel	Master	Core PCSM signals
core_powerdown_out[26]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[26]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[26]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error

Name	Protocol	Type	Description
cryptodisable[26]	Signal	Slave	Disable cryptography extensions after reset.
cti[26]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[26]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[26]	Signal	Master	No power-down request.
dbgpwrupreq[26]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[26]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[26]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[26]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[26]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[26]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[26]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[26]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[26]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[26]	Value_64	Slave	Reset vector base address.

Name	Protocol	Type	Description
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[26]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[26]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[26]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[26]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[26]	Signal	Slave	Virtualised FIQ.
virq[26]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[26]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA520CT_CortexA720CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance

of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_complex_mapping

Defines Complex descriptions for platforms that support several Cores per Complex like Cortex-A510. JSON format: {"complex0": { "cores" : [0, 1], "l2-cache" : {"exists":1, "size":16MB}}, ... , "complexN": { "cores" : [<core_list>], "l2-cache" : {"exists":1, "size":16MB}}} where <core_list> is the list of cores in the complexN. Effective only when the parameter value is not empty.

Type

string

Default value

```
{"complex0": { "cores": [0, 1], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex1":  
{ "cores": [2, 3], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex2": { "cores": [4, 5], "l2-  
cache" : {"exists":1, "size":"16MB"}}, "complex3": { "cores": [6, 7], "l2-cache" : {"exists":1,  
"size":"16MB"}}, "complex4": { "cores": [8, 9], "l2-cache" : {"exists":1, "size":"16MB"}},  
"complex5": { "cores": [10, 11], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex6":  
{ "cores": [12, 13], "l2-cache" : {"exists":1, "size":"16MB"} }
```

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_bus_width_in_bytes

L3 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_bus_width_in_bytes

L3 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantums in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

subcluster0.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster0.core_cache_protection

core_cache_protection can change ERROFR, ERROPFGF and ERROPFGCTL fields. Possible values are: -1:Not implemented (by default), 0:Disabled, 1:Enabled

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster0.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster0.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster0.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster0.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster0.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster0.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster0.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster0.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster0.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster0.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster0.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster0.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance

of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster0.has_actlr2

If true ACTLR2 exists and ACTLR2(NS) is aliased to ACTLR_EL1[63:32]

Type

bool

Default value

0x1

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster0.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.ras_extra_configurations

Miscellaneous configurations for error records. An array of JSON objects. Note for ERXCTLR_EL1 register it only allows to define the mask value for the IMPDEF fields, ie bits [63:32] and bit 1, but its reset value applies on all fields. Note for ERXMISCn masks - these are 64 bit masks covering the 64 bit registers ERXMISCn_EL1. E.g. [{"Index": 0, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXMISC1_mask": 0x0, "ERXMISC1_reset": 0x0, "ERXMISC2_mask": 0x0, "ERXMISC2_reset": 0x0, "ERXMISC3_mask": 0x0, "ERXMISC3_reset": 0x0, "ERXCTLR_EL1_mask": 0x0, "ERXCTLR_EL1_reset": 0x0}, {"Index": 1, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXSTATUS_IERR_mask": 0x300}]

Type

string

Default value

```
[ { "Index": 1, "EXMISCO_mask": 0xFFFFc0003fc3, "EXMISC1_mask":
0x03F870003FF30f07, "ERXPFGCTL_reset": 0x1000 }, { "Index":
2, "EXMISCO_mask": 0xFFFFe007ffc0, "EXMISCO_reset": 0x2,
"ERXSTATUS_IERR_mask": 0x300 , "EXMISC1_mask": 0x0FF8700fFFF31f0f,
"ERXPFGCTL_reset": 0x1000 } ]
```

subcluster0.ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

subcluster0.revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster1.ete.ETE_REVISION

ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3

Type

int

Default value

0x1

subcluster1.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster1.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster1.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster1.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster1.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster1.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster1.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster1.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster1.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster1.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster1.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster1.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster1.ete.TSMARK

Whether timestamp markers are supported

Type

bool

Default value

0x1

subcluster1.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster1.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster1.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster1.has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster1.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster1.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster1.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlbi invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.41 ARMCortexA520CT_CortexA720CT_CortexX4CT

ARMCortexA520CT_CortexA720CT_CortexX4CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-345: IP revisions support

Revision	Quality level
CortexA520 r0p1	Full support
CortexA720 r0p1	Full support
CortexX4 r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Table 3-346: Model quality changes

From	To
CortexA520 r0p0=rel	CortexA520 r0p1=rel
Hunter r0p0=pre	CortexA720 r0p1=rel
CortexX4 r0p0=pre	CortexX4 r0p0=pre

Table 3-347: IP revision changes

From	To
CortexA520 r0p1	CortexA520 r0p1
CortexA720 r0p1	CortexA720 r0p1
CortexX4 r0p0	CortexX4 r0p0

Ports added:

- cluster_pcs_m_pchannel
- core_pcs_m_pchannel

Parameters added:

- core_complex_mapping
- mpmm_accumulator_multiplier
- subcluster0.tlbi_stall_enabled
- subcluster1.tlbi_stall_enabled
- subcluster2.tlbi_stall_enabled

Iris and MTI instances for ARMCortexA520CT_CortexA720CT_CortexX4CT

This model has the following Iris instances:

Table 3-348: ARMCortexA520CT_CortexA720CT_CortexX4CT Iris instances

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT_CortexX4CT	Cluster_ARM_CortexA520_CortexA720_CortexX4_Heterogeneous
ARMCortexA520CT_CortexA720CT_CortexX4CT.AMU	PVBusLogger
ARMCortexA520CT_CortexA720CT_CortexX4CT.AMU.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT_CortexX4CT.DAP	PVBusLogger
ARMCortexA520CT_CortexA720CT_CortexX4CT.DAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU	DSU
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_cluster	PPUv1

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core0	PPUv1
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core1	PPUv1
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core2	PPUv1
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core2.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[7]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[8]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.MMAP	PVBusLogger
ARMCortexA520CT_CortexA720CT_CortexX4CT.MMAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT_CortexX4CT.ext_bus	PVBusLogger
ARMCortexA520CT_CortexA720CT_CortexX4CT.ext_bus.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT_CortexX4CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0	Subcluster_ARM_Cortex-A520
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0	ARM_Cortex-A520
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1	Subcluster_ARM_Cortex-A720
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0	ARM_Cortex-A720
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu1.dtlb	TlbCadi

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2	Subcluster_ARM_Cortex-X4
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0	ARM_Cortex-X4
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l1licache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu2.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-349: ARMCortexA520CT_CortexA720CT_CortexX4CT MTI instances

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT_CortexX4CT	ARMv8Cluster
ARMCortexA520CT_CortexA720CT_CortexX4CT.AMU	PVBusLogger
ARMCortexA520CT_CortexA720CT_CortexX4CT.AMU.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT_CortexX4CT.DAP	PVBusLogger
ARMCortexA520CT_CortexA720CT_CortexX4CT.DAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU	DSU
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_cluster	PPUv1
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core0	PPUv1
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core1	PPUv1
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core2	PPUv1
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.PPU_core2.busslave	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.mpam_busslave	PVBusSlave

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[7]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.shared_cache.upstream[8]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.MMAP	PVBusLogger
ARMCortexA520CT_CortexA720CT_CortexX4CT.MMAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT_CortexX4CT.ext_bus	PVBusLogger
ARMCortexA520CT_CortexA720CT_CortexX4CT.ext_bus.mapper	PVBusMapper
ARMCortexA520CT_CortexA720CT_CortexX4CT.gic_cpuif_decoder_cluster	GiCv3CPUInterfaceDecoder
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0	ARM_Cortex-A520
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0	ARM_Cortex-A720
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l1licache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0	ARM_Cortex-X4
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l1licache	PVCache

InstanceName	ComponentName
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA720CT_CortexX4CT.subcluster2.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA520CT_CortexA720CT_CortexX4CT contains the following CADI targets:

- ARM_Cortex-A520
- ARM_Cortex-A720
- ARM_Cortex-X4
- Cluster_ARM_CortexA520_CortexA720_CortexX4_Heterogeneous
- PVCache
- Subcluster_ARM_Cortex-A520
- Subcluster_ARM_Cortex-A720
- Subcluster_ARM_Cortex-X4
- TlbCadi

About ARMCortexA520CT_CortexA720CT_CortexX4CT

The number of cores in each subcluster is configurable using the following parameters:

subcluster0.NUM_CORES

Possible values are 1-12 (ARMCortexA520CT).

subcluster1.NUM_CORES

Possible values are 1-12 (ARMCortexA720CT).

subcluster2.NUM_CORES

Possible values are 1-12 (ARMCortexX4CT).

The total number of cores in the cluster cannot exceed 14.

Port arrays are expanded to the width needed for the maximum number of cores in each subcluster. Use the following port array indexes:

- `<port_name>[0-11]` for cores in subcluster0.
- `<port_name>[12-23]` for cores in subcluster1.
- `<port_name>[24-35]` for cores in subcluster2.



Note

All instances in the Master cross trigger matrix port array, `cti[36]` must be connected, regardless of the `NUM_CORES` value used.

Core-specific parameters have the following prefixes:

- `subcluster0.cpu0` to `subcluster0.cpu11` identify cores in `subcluster0`.
- `subcluster1.cpu0` to `subcluster1.cpu11` identify cores in `subcluster1`.
- `subcluster2.cpu0` to `subcluster2.cpu11` identify cores in `subcluster2`.

For information about the cores in this model, see:

- [3.5.39 ARMCortexA520CT](#) on page 1703.
- [3.5.47 ARMCortexA720CT](#) on page 2054.
- [3.5.73 ARMCortexX4CT](#) on page 2551.

Ports for ARMCortexA520CT_CortexA720CT_CortexX4CT

Table 3-350: Ports

Name	Protocol	Type	Description
<code>acp_s[2]</code>	PVBus	Slave	AXI ACP slave port.
<code>AENDOMP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND1MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND2MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND3MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTOMP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART1MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART2MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART3MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[36]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
<code>cluster_pcsn_pchannel</code>	PChannel	Master	Cluster PCSM signal
<code>cluster_powerdown_out</code>	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the cluster power info

Name	Protocol	Type	Description
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[36]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[36]	Signal	Master	Timer signals to SOC
CNTHVIRQ[36]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[36]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[36]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[36]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[36]	Signal	Master	Timer signals to SOC.
commirq[36]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[36]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[36]	PChannel	Master	Core PCSM signals
core_powerdown_out[36]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[36]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[36]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[36]	Signal	Slave	Disable cryptography extensions after reset.
cti[36]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[36]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[36]	Signal	Master	No power-down request.
dbgpwrupreq[36]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[36]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components

Name	Protocol	Type	Description
gicv3_redistributor_s[36]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[36]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[36]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[36]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[36]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[36]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[36]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[36]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[36]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[36]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[36]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[36]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[36]	Signal	Slave	Virtualised FIQ.
virq[36]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[36]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A520CT_Cortex A720CT_Cortex X4CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAIN

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_complex_mapping

Defines Complex descriptions for platforms that support several Cores per Complex like Cortex-A510. JSON format: {"complex0": { "cores" : [0, 1], "l2-cache" : {"exists":1, "size":16MB}}, ... , "complexN": { "cores" : [<core_list>], "l2-cache" : {"exists":1, "size":16MB}}} where <core_list> is the list of cores in the complexN. Effective only when the parameter value is not empty.

Type

string

Default value

```
{"complex0": { "cores": [0, 1], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex1":
{ "cores": [2, 3], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex2": { "cores": [4, 5], "l2-
cache" : {"exists":1, "size":"16MB"}}, "complex3": { "cores": [6, 7], "l2-cache" : {"exists":1,
"size":"16MB"}}, "complex4": { "cores": [8, 9], "l2-cache" : {"exists":1, "size":"16MB"}},
"complex5": { "cores": [10, 11], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex6":
{ "cores": [12, 13], "l2-cache" : {"exists":1, "size":"16MB"}} }
```


core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_bus_width_in_bytes

L3 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_bus_width_in_bytes

L3 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

subcluster0.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster0.core_cache_protection

core_cache_protection can change ERROFR, ERROPFGF and ERROPFGCTL fields. Possible values are: -1:Not implemented (by default), 0:Disabled, 1:Enabled

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `l2cache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster0.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster0.ete.PIDR_CMOD

TRCPIDR CMOD value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster0.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster0.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster0.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster0.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster0.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster0.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster0.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster0.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster0.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster0.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster0.has_actlr2

If true ACLTR2 exists and ACTLR2(NS) is aliased to ACTLR_EL1[63:32]

Type

bool

Default value

0x1

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster0.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.ras_extra_configurations

Miscellaneous configurations for error records. An array of JSON objects. Note for ERXCTLR_EL1 register it only allows to define the mask value for the IMPDEF fields, ie bits [63:32] and bit 1, but its reset value applies on all fields. Note for ERXMISCn masks - these are 64 bit masks covering the 64 bit registers ERXMISCn_EL1. E.g. [{"Index": 0, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXMISC1_mask": 0x0, "ERXMISC1_reset": 0x0, "ERXMISC2_mask": 0x0, "ERXMISC2_reset": 0x0, "ERXMISC3_mask": 0x0, "ERXMISC3_reset": 0x0, "ERXCTLR_EL1_mask": 0x0, "ERXCTLR_EL1_reset": 0x0}, {"Index": 1, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXSTATUS_IERR_mask": 0x300}]

Type

string

Default value

```
[ { "Index": 1, "ERXMISCO_mask": 0xFFFFc0003fc3, "ERXMISC1_mask":
0x03F870003FF30f07, "ERXPFGCTL_reset": 0x1000 }, { "Index":
2, "ERXMISCO_mask": 0xFFFFe007ffc0, "ERXMISCO_reset": 0x2,
"ERXSTATUS_IERR_mask": 0x300 , "ERXMISC1_mask": 0x0FF8700ffff31f0f,
"ERXPFGCTL_reset": 0x1000 } ]
```

subcluster0.ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

subcluster0.revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value

subcluster1.cpuX.semihosting-cwd

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster1.ete.ETE_REVISION

ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3

Type

int

Default value

0x1

subcluster1.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster1.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster1.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster1.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster1.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster1.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster1.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster1.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster1.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster1.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster1.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster1.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster1.ete.TSMARK

Whether timestamp markers are supported

Type

bool

Default value

0x1

subcluster1.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect.
1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster1.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster1.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster1.has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster1.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster1.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster1.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster1.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC*

instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster2.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster2.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster2.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster2.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster2.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster2.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster2.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster2.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster2.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster2.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster2.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster2.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster2.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster2.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster2.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster2.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster2.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster2.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster2.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster2.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster2.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster2.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster2.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster2.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster2.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster2.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster2.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster2.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster2.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster2.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.ecv_support_level

Implement Enhanced Counter Virtualization feature from ARMv8.6. 0, Not supported. 1, fully supported without CNTPOFF. 2, fully supported with CNTPOFF (FEAT_ECV).

Type

int

Default value

0x2

subcluster2.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster2.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster2.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster2.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster2.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster2.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster2.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster2.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster2.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster2.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster2.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster2.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster2.ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO , 4 = UER, 5 = CE.

Type

int

Default value

0x2

subcluster2.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster2.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster2.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect.
1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster2.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster2.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster2.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

subcluster2.has_v8_7_spe_inverted_filtering

Where FEAT_SPEv1p2 is implemented, whether the inverting filtering by events

Type

bool

Default value

0x0

subcluster2.has_v8_7_spe_previous_branch_target

Where FEAT_SPEv1p2 is implemented, whether the optional branch target feature is implemented

Type

bool

Default value

0x0

subcluster2.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster2.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster2.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster2.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster2.mpam_has_altsp

MPAM Whether MPAMIDR_EL1.HAS_ALTSP bit is set or clear

Type

bool

Default value

0x0

subcluster2.mpamidr_has_force_ns

Whether MPAMIDR_EL1.HAS_FORCE_NS bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster2.mpamidr_has_sdeflt

Whether MPAMIDR_EL1.HAS_SDEFLT bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

subcluster2.mpamidr_has_tidr

Whether MPAMIDR_EL1.HAS_TIDR bit is set or clear

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

subcluster2.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x1f

subcluster2.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster2.tcr_txsz_undersize_should_fault

If large VA is not supported, Whether undersized TxSZ value should generate translation fault

Type

bool

Default value

0x0

subcluster2.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster2.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster2.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.42 **ARMCortexA520CT_CortexA725CT**

ARMCortexA520CT_CortexA725CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-351: IP revisions support

Revision	Quality level
CortexA520 r0p1	Full support
CortexA725 r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA520CT_CortexA725CT

This model has the following Iris instances:

Table 3-352: ARMCortexA520CT_CortexA725CT Iris instances

InstanceName	ComponentName
ARMCortexA520CT_CortexA725CT	Cluster_ARM_Cortex-A520_CortexA725_Heterogeneous
ARMCortexA520CT_CortexA725CT.AMU	PVBusLogger

InstanceName	ComponentName
ARMCortexA520CT_CortexA725CT.AMU.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT.DAP	PVBusLogger
ARMCortexA520CT_CortexA725CT.DAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT.DSU	DSU
ARMCortexA520CT_CortexA725CT.DSU.PPU_cluster	PPUv1
ARMCortexA520CT_CortexA725CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.PPU_core0	PPUv1
ARMCortexA520CT_CortexA725CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.PPU_core1	PPUv1
ARMCortexA520CT_CortexA725CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA520CT_CortexA725CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache	PVCache
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.MMAP	PVBusLogger
ARMCortexA520CT_CortexA725CT.MMAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT.ext_bus	PVBusLogger
ARMCortexA520CT_CortexA725CT.ext_bus.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA520CT_CortexA725CT.subcluster0	Subcluster_ARM_Cortex-A520
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0	ARM_Cortex-A520
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster1	Subcluster_ARM_Cortex-A725
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0	ARM_Cortex-A725
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster1.cpu1.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-353: ARMCortexA520CT_CortexA725CT MTI instances

InstanceName	ComponentName
ARMCortexA520CT_CortexA725CT	ARMv8Cluster
ARMCortexA520CT_CortexA725CT.AMU	PVBusLogger
ARMCortexA520CT_CortexA725CT.AMU.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT.DAP	PVBusLogger
ARMCortexA520CT_CortexA725CT.DAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT.DSU	DSU
ARMCortexA520CT_CortexA725CT.DSU.PPU_cluster	PPUv1
ARMCortexA520CT_CortexA725CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.PPU_core0	PPUv1
ARMCortexA520CT_CortexA725CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.PPU_core1	PPUv1
ARMCortexA520CT_CortexA725CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA520CT_CortexA725CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache	PVCache
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[3]	PVBusSlave

InstanceName	ComponentName
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA520CT_CortexA725CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.MMAP	PVBusLogger
ARMCortexA520CT_CortexA725CT.MMAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT.ext_bus	PVBusLogger
ARMCortexA520CT_CortexA725CT.ext_bus.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0	ARM_Cortex-A520
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0	ARM_Cortex-A725
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l1licache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA520CT_CortexA725CT contains the following CADI targets:

- ARM_Cortex-A520
- ARM_Cortex-A725
- Cluster_ARM_Cortex-A520_CortexA725_Heterogeneous
- PVCache
- Subcluster_ARM_Cortex-A520
- Subcluster_ARM_Cortex-A725
- TlbCadi

Ports for ARMCortexA520CT_CortexA725CT

Table 3-354: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[26]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[26]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[26]	Signal	Master	Timer signals to SOC
CNTHVIRQ[26]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[26]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[26]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[26]	Signal	Master	Timer signals to SOC.

Name	Protocol	Type	Description
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[26]	Signal	Master	Timer signals to SOC.
commirq[26]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[26]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[26]	PChannel	Master	Core PCSM signals
core_powerdown_out[26]	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the core power info
coreerrirq[26]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[26]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[26]	Signal	Slave	Disable cryptography extensions after reset.
cti[26]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[26]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[26]	Signal	Master	No power-down request.
dbgpwrupreq[26]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[26]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPUs that can be used to reset GICCLK domain components
gicv3_redistributor_s[26]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[26]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[26]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[26]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.

Name	Protocol	Type	Description
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[26]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[26]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[26]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[26]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[26]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[26]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[26]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[26]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[26]	Signal	Slave	Virtualised FIQ.
virq[26]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[26]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA520CT_CortexA725CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_complex_mapping

Defines Complex descriptions for platforms that support several Cores per Complex like Cortex-A510. JSON format: {"complex0": { "cores" : [0, 1], "l2-cache" : {"exists":1, "size":16MB}}, ... , "complexN": { "cores" : [<core_list>], "l2-cache" : {"exists":1, "size":16MB}}} where <core_list> is the list of cores in the complexN. Effective only when the parameter value is not empty.

Type

string

Default value

```
{"complex0": { "cores": [0, 1], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex1":
{ "cores": [2, 3], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex2": { "cores": [4, 5], "l2-
cache" : {"exists":1, "size":"16MB"}}, "complex3": { "cores": [6, 7], "l2-cache" : {"exists":1,
"size":"16MB"}}, "complex4": { "cores": [8, 9], "l2-cache" : {"exists":1, "size":"16MB"}},
"complex5": { "cores": [10, 11], "l2-cache" : {"exists":1, "size":"16MB"}}, "complex6":
{ "cores": [12, 13], "l2-cache" : {"exists":1, "size":"16MB"} } }
```

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_bus_width_in_bytes

L3 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_bus_width_in_bytes

L3 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

subcluster0.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster0.core_cache_protection

core_cache_protection can change ERROFR, ERROPFGF and ERROPFGCTL fields. Possible values are: -1:Not implemented (by default), 0:Disabled, 1:Enabled

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster0.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster0.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster0.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster0.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster0.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster0.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster0.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster0.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster0.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster0.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster0.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster0.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect.
 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster0.has_actlr2

If true ACLTR2 exists and ACTLR2(NS) is aliased to ACTLR_EL1[63:32]

Type

bool

Default value

0x1

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster0.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.ras_extra_configurations

Miscellaneous configurations for error records. An array of JSON objects. Note for ERXCTLR_EL1 register it only allows to define the mask value for the IMPDEF fields, ie bits [63:32] and bit 1, but its reset value applies on all fields. Note for ERXMISCN masks - these are 64 bit masks covering the 64 bit registers ERXMISCN_EL1. E.g. [{"Index": 0, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXMISC1_mask": 0x0, "ERXMISC1_reset": 0x0, "ERXMISC2_mask": 0x0, "ERXMISC2_reset": 0x0, "ERXMISC3_mask": 0x0, "ERXMISC3_reset": 0x0, "ERXCTLR_EL1_mask": 0x0, "ERXCTLR_EL1_reset": 0x0}, {"Index": 1, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXSTATUS_IERR_mask": 0x300}]

Type

string

Default value

```
[ { "Index": 1, "ERXMISCO_mask": 0xFFFFc0003fc3, "ERXMISC1_mask":
0x03F870003FF30f07, "ERXPFGCTL_reset": 0x1000 }, { "Index":
2, "ERXMISCO_mask": 0xFFFFe007ffc0, "ERXMISCO_reset": 0x2,
"ERXSTATUS_IERR_mask": 0x300 , "ERXMISC1_mask": 0x0FF8700fFFF31f0f,
"ERXPFGCTL_reset": 0x1000 } ]
```

subcluster0.ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

subcluster0.revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x100000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster1.ete.ETE_REVISION

ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3

Type

int

Default value

0x1

subcluster1.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster1.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster1.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster1.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster1.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster1.ete.RETSTACK

Return stack depth

Type

int

Default value

0x1

subcluster1.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster1.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster1.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster1.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster1.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value

subcluster1.ete.TRCRSRTA_FORCED_EXCEP

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster1.ete.TSMARK

Whether timestamp markers are supported

Type

bool

Default value

0x1

subcluster1.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster1.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster1.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster1.has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster1.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster1.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster1.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster1.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.43 ARM CortexA520CT_CortexA725CT_CortexX925CT

ARM CortexA520CT_CortexA725CT_CortexX925CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-355: IP revisions support

Revision	Quality level
CortexA520 r0p1	Full support
CortexA725 r0p0	Preliminary support
CortexX925 r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARM CortexA520CT_CortexA725CT_CortexX925CT

This model has the following Iris instances:

Table 3-356: ARM CortexA520CT_CortexA725CT_CortexX925CT Iris instances

InstanceName	ComponentName
ARM CortexA520CT_CortexA725CT_CortexX925CT	Cluster_ARM_Cortex-A520_CortexA725_CortexX925_Heterogeneous
ARM CortexA520CT_CortexA725CT_CortexX925CT.AMU	PVBusLogger
ARM CortexA520CT_CortexA725CT_CortexX925CT.AMU.mapper	PVBusMapper
ARM CortexA520CT_CortexA725CT_CortexX925CT.DAP	PVBusLogger
ARM CortexA520CT_CortexA725CT_CortexX925CT.DAP.mapper	PVBusMapper
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU	DSU
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_cluster	PPUv1
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_cluster.busslave	PVBusSlave
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core0	PPUv1
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core0.busslave	PVBusSlave
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core1	PPUv1
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core1.busslave	PVBusSlave
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core2	PPUv1
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core2.busslave	PVBusSlave
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.mpam_busslave	PVBusSlave
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache	PVCache
ARM CortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[7]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[8]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.MMAP	PVBusLogger
ARMCortexA520CT_CortexA725CT_CortexX925CT.MMAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT_CortexX925CT.ext_bus	PVBusLogger
ARMCortexA520CT_CortexA725CT_CortexX925CT.ext_bus.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT_CortexX925CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0	Subcluster_ARM_Cortex-A520
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0	ARM_Cortex-A520
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l1lcache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l1lcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1	Subcluster_ARM_Cortex-A725

InstanceName	ComponentName
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0	ARM_Cortex-A725
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu1.dtlb	TlbCadi
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2	Subcluster_ARM_Cortex-X925
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0	ARM_Cortex-X925
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l1icache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu2.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-357: ARMCortexA520CT_CortexA725CT_CortexX925CT MTI instances

InstanceName	ComponentName
ARMCortexA520CT_CortexA725CT_CortexX925CT	ARMv8Cluster

InstanceName	ComponentName
ARMCortexA520CT_CortexA725CT_CortexX925CT.AMU	PVBusLogger
ARMCortexA520CT_CortexA725CT_CortexX925CT.AMU.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT_CortexX925CT.DAP	PVBusLogger
ARMCortexA520CT_CortexA725CT_CortexX925CT.DAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU	DSU
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_cluster	PPUv1
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core0	PPUv1
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core1	PPUv1
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core2	PPUv1
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.PPU_core2.busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[7]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.shared_cache.upstream[8]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.MMAP	PVBusLogger
ARMCortexA520CT_CortexA725CT_CortexX925CT.MMAP.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT_CortexX925CT.ext_bus	PVBusLogger
ARMCortexA520CT_CortexA725CT_CortexX925CT.ext_bus.mapper	PVBusMapper
ARMCortexA520CT_CortexA725CT_CortexX925CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0	ARM_Cortex-A520
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l1licache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache	PVCache

InstanceName	ComponentName
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0	ARM_Cortex-A725
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0	ARM_Cortex-X925
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.UTLB	TLB
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l1dcache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l1icache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l2cache	PVCache
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA520CT_CortexA725CT_CortexX925CT.subcluster2.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA520CT_CortexA725CT_CortexX925CT contains the following CADI targets:

- ARM_Cortex-A520
- ARM_Cortex-A725
- ARM_Cortex-X925
- Cluster_ARM_Cortex-A520_CortexA725_CortexX925_Heterogeneous
- PVCache
- Subcluster_ARM_Cortex-A520
- Subcluster_ARM_Cortex-A725
- Subcluster_ARM_Cortex-X925
- TlbCadi

Ports for ARMCortexA520CT_CortexA725CT_CortexX925CT

Table 3-358: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[36]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcsn_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[36]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[36]	Signal	Master	Timer signals to SOC
CNTHVIRQ[36]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[36]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[36]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[36]	Signal	Master	Timer signals to SOC.

Name	Protocol	Type	Description
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[36]	Signal	Master	Timer signals to SOC.
commirq[36]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[36]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[36]	PChannel	Master	Core PCSM signals
core_powerdown_out[36]	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the core power info
coreerrirq[36]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[36]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[36]	Signal	Slave	Disable cryptography extensions after reset.
cti[36]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[36]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[36]	Signal	Master	No power-down request.
dbgpwrupreq[36]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[36]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPUs that can be used to reset GICCLK domain components
gicv3_redistributor_s[36]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[36]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[36]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[36]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.

Name	Protocol	Type	Description
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[36]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[36]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[36]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[36]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[36]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[36]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[36]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[36]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[36]	Signal	Slave	Virtualised FIQ.
virq[36]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[36]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA520CT_CortexA725CT_CortexX925CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

subcluster0.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster0.core_cache_protection

core_cache_protection can change ERROFR, ERROPFGF and ERROPFGCTL fields. Possible values are: -1:Not implemented (by default), 0:Disabled, 1:Enabled

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_bus_width_in_bytes

L2 Cache read bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x10

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_bus_width_in_bytes

L2 Cache write bus width in bytes used to calculate per-access timing annotations

Type

int

Default value

0x20

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster0.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster0.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster0.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster0.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster0.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster0.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster0.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster0.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster0.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster0.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster0.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster0.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect.
1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster0.has_actlr2

If true ACLTR2 exists and ACTLR2(NS) is aliased to ACTLR_EL1[63:32]

Type

bool

Default value

0x1

subcluster0.has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

subcluster0.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster0.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.ras_extra_configurations

Miscellaneous configurations for error records. An array of JSON objects. Note for ERXCTLR_EL1 register it only allows to define the mask value for the IMPDEF fields, ie bits [63:32] and bit 1, but its reset value applies on all fields. Note for ERXMISCN masks - these are 64 bit masks covering the 64 bit registers ERXMISCN_EL1. E.g. [{"Index": 0, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXMISC1_mask": 0x0, "ERXMISC1_reset": 0x0, "ERXMISC2_mask": 0x0, "ERXMISC2_reset": 0x0, "ERXMISC3_mask": 0x0, "ERXMISC3_reset": 0x0, "ERXCTLR_EL1_mask": 0x0, "ERXCTLR_EL1_reset": 0x0}, {"Index": 1, "ERXMISCO_mask": 0x0, "ERXMISCO_reset": 0x0, "ERXSTATUS_IERR_mask": 0x300}]

Type

string

Default value

```
[ { "Index": 1, "ERXMISCO_mask": 0xFFFFc0003fc3, "ERXMISC1_mask":
0x03F870003FF30f07, "ERXPFGCTL_reset": 0x1000 }, { "Index":
2, "ERXMISCO_mask": 0xFFFFe007ffc0, "ERXMISCO_reset": 0x2,
"ERXSTATUS_IERR_mask": 0x300 , "ERXMISC1_mask": 0x0FF8700fFFF31f0f,
"ERXPFGCTL_reset": 0x1000 } ]
```

subcluster0.ras_pfg_clock_mhz

RAS Pseudo-Fault generation clock rate in MHz

Type

int

Default value

0xc

subcluster0.revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x0

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x100000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster1.ete.ETE_REVISION

ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3

Type

int

Default value

0x1

subcluster1.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster1.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster1.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster1.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster1.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster1.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster1.ete.RETSTACK

Return stack depth

Type

int

Default value

0x1

subcluster1.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster1.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster1.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster1.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster1.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value

subcluster1.ete.TRCRSRTA_FORCED_EXCEP

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster1.ete.TSMARK

Whether timestamp markers are supported

Type

bool

Default value

0x1

subcluster1.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster1.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster1.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster1.has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster1.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster1.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster1.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster1.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster2.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster2.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster2.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster2.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster2.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster2.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster2.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster2.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster2.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster2.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster2.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster2.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster2.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster2.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster2.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster2.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster2.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster2.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster2.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster2.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster2.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster2.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster2.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster2.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster2.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster2.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster2.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster2.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster2.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster2.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.ecv_support_level

Implement Enhanced Counter Virtualization feature from ARMv8.6. 0, Not supported. 1, fully supported without CNTPOFF. 2, fully supported with CNTPOFF (FEAT_ECV).

Type

int

Default value

0x2

subcluster2.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster2.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster2.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster2.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster2.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster2.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster2.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster2.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster2.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster2.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster2.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster2.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster2.ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x2

subcluster2.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster2.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster2.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster2.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster2.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster2.has_mt_pmu_disable_feature

Implements Multi-threading PMU disable extension from ARMv8.6 (FEAT_MTPMU). 0: FEAT_MTPMU is disabled, 1: FEAT_MTPMU is enabled if ARMv8.6 is implemented, 2: FEAT_MTPMU is cherry-picked, 0xF: The feature is disabled and is represented by value 0xF in ID_AA64DFR0_EL1.MTPMU.

Type

int

Default value

0x0

subcluster2.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

subcluster2.has_v8_7_spe_inverted_filtering

Where FEAT_SPEv1p2 is implemented, whether the inverting filtering by events

Type

bool

Default value

0x0

subcluster2.has_v8_7_spe_previous_branch_target

Where FEAT_SPEv1p2 is implemented, whether the optional branch target feature is implemented

Type

bool

Default value

0x0

subcluster2.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster2.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster2.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster2.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster2.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster2.mpam_has_altsp

MPAM Whether MPAMIDR_EL1.HAS_ALTSP bit is set or clear

Type

bool

Default value

0x0

subcluster2.mpamidr_has_force_ns

Whether MPAMIDR_EL1.HAS_FORCE_NS bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster2.mpamidr_has_sdeflt

Whether MPAMIDR_EL1.HAS_SDEFLT bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

subcluster2.mpamidr_has_tidr

Whether MPAMIDR_EL1.HAS_TIDR bit is set or clear

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

subcluster2.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x1f

subcluster2.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster2.tcr_txsz_undersize_should_fault

If large VA is not supported, Whether undersized TxSZ value should generate translation fault

Type

bool

Default value

0x0

subcluster2.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster2.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster2.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster2.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.44 ARMCortexA710CT

ARMCortexA710CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-359: IP revisions support

Revision	Quality level
r2p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `mpmm_accumulator_multiplier`
- `tlbi_stall_enabled`

Iris and MTI instances for ARMCortexA710CT

This model has the following Iris instances:

Table 3-360: ARMCortexA710CT Iris instances

InstanceName	ComponentName
ARMCortexA710CT	Cluster_ARM_Cortex-A710
ARMCortexA710CT.AMU	PVBusLogger
ARMCortexA710CT.AMU.mapper	PVBusMapper
ARMCortexA710CT.DAP	PVBusLogger
ARMCortexA710CT.DAP.mapper	PVBusMapper
ARMCortexA710CT.DSU	DSU
ARMCortexA710CT.DSU.PPU_cluster	PPUv1
ARMCortexA710CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA710CT.DSU.PPU_core0	PPUv1
ARMCortexA710CT.DSU.PPU_core0.busslave	PVBusSlave

InstanceName	ComponentName
ARMCortexA710CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA710CT.DSU.shared_cache	PVCache
ARMCortexA710CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA710CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA710CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA710CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA710CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA710CT.MMAP	PVBusLogger
ARMCortexA710CT.MMAP.mapper	PVBusMapper
ARMCortexA710CT.cpu0	ARM_Cortex-A710
ARMCortexA710CT.cpu0.UTLB	TLB
ARMCortexA710CT.cpu0.dtlb	TlbCadi
ARMCortexA710CT.cpu0.l1dcache	PVCache
ARMCortexA710CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA710CT.cpu0.l1icache	PVCache
ARMCortexA710CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA710CT.cpu0.l2cache	PVCache
ARMCortexA710CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA710CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA710CT.ext_bus	PVBusLogger
ARMCortexA710CT.ext_bus.mapper	PVBusMapper
ARMCortexA710CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-361: ARMCortexA710CT MTI instances

InstanceName	ComponentName
ARMCortexA710CT	ARMv8Cluster
ARMCortexA710CT.AMU	PVBusLogger
ARMCortexA710CT.AMU.mapper	PVBusMapper
ARMCortexA710CT.DAP	PVBusLogger
ARMCortexA710CT.DAP.mapper	PVBusMapper
ARMCortexA710CT.DSU	DSU
ARMCortexA710CT.DSU.PPU_cluster	PPUv1
ARMCortexA710CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA710CT.DSU.PPU_core0	PPUv1
ARMCortexA710CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA710CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA710CT.DSU.shared_cache	PVCache
ARMCortexA710CT.DSU.shared_cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA710CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA710CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA710CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA710CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA710CT.MMAP	PVBusLogger
ARMCortexA710CT.MMAP.mapper	PVBusMapper
ARMCortexA710CT.cpu0	ARM_Cortex-A710
ARMCortexA710CT.cpu0.UTLB	TLB
ARMCortexA710CT.cpu0.l1dcache	PVCache
ARMCortexA710CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA710CT.cpu0.l1icache	PVCache
ARMCortexA710CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA710CT.cpu0.l2cache	PVCache
ARMCortexA710CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA710CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA710CT.ext_bus	PVBusLogger
ARMCortexA710CT.ext_bus.mapper	PVBusMapper
ARMCortexA710CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA710CT contains the following CADI targets:

- ARM_Cortex-A710
- Cluster_ARM_Cortex-A710
- PVCache
- TlbCadi

About ARMCortexA710CT

The model supports the following features:

- DynamIQ™ Shared Unit-110 (DSU-110) system registers.
- L2 cache is supported at the per-core level only and there is no implementation of Core-Complex with shared L2 cache yet.
- PChannel for the cluster and for each core.
- BROADCASTPERSIST pin.
- Optional peripheral port.
- L3Cache partition.
- Per-core clock.
- Utility bus.
- AArch32 at ELO.

Support for the following features is planned for a future release:

- Transporting architectural metadata tags over TLM2 (OSCI) and AMBA-PV (the Arm TLM2 extensions) bus protocols.
- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, and nPMBIRQ signals.

The following features will not be implemented:

- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- Error correction or detection features.
- Self-test features (MBIST).
- Snoop filtering.

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (crypto.dll or crypto.so) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA710CT

Table 3-362: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.

Name	Protocol	Type	Description
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[12]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[12]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[12]	Signal	Master	Timer signals to SOC
CNTHVIRQ[12]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[12]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[12]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[12]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[12]	Signal	Master	Timer signals to SOC.
commirq[12]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[12]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[12]	PChannel	Master	Core PCSM signals
core_powerdown_out[12]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[12]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[12]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[12]	Signal	Slave	Disable cryptography extensions after reset.
cti[12]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[12]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[12]	Signal	Master	No power-down request.

Name	Protocol	Type	Description
dbgpwrupreq[12]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[12]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[12]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[12]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmuirq[12]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[12]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[12]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[12]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[12]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[12]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[12]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[12]	Signal	Master	Interrupt signal from the trace buffer unit.

Name	Protocol	Type	Description
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[12]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[12]	Signal	Slave	Virtualised FIQ.
virq[12]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[12]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA710CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when `dcache-state_modelled=true`

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size `dcache-read_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-read_latency` is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. `dcache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x20

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x1

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x2

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x0

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).

Type

int

Default value

0x2

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases

where execution of quantums in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.45 ARMCortexA715CT

ARMCortexA715CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-363: IP revisions support

Revision	Quality level
r1p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `mpmm_accumulator_multiplier`
- `tlbi_stall_enabled`

Iris and MTI instances for ARM Cortex-A715CT

This model has the following Iris instances:

Table 3-364: ARM Cortex-A715CT Iris instances

InstanceName	ComponentName
ARM Cortex-A715CT	Cluster_ARM_Cortex-A715
ARM Cortex-A715CT.AMU	PVBusLogger
ARM Cortex-A715CT.AMU.mapper	PVBusMapper
ARM Cortex-A715CT.DAP	PVBusLogger
ARM Cortex-A715CT.DAP.mapper	PVBusMapper
ARM Cortex-A715CT.DSU	DSU
ARM Cortex-A715CT.DSU.PPU_cluster	PPUv1
ARM Cortex-A715CT.DSU.PPU_cluster.busslave	PVBusSlave
ARM Cortex-A715CT.DSU.PPU_core0	PPUv1
ARM Cortex-A715CT.DSU.PPU_core0.busslave	PVBusSlave
ARM Cortex-A715CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM Cortex-A715CT.DSU.mpam_busslave	PVBusSlave
ARM Cortex-A715CT.DSU.shared_cache	PVCache
ARM Cortex-A715CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM Cortex-A715CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARM Cortex-A715CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARM Cortex-A715CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARM Cortex-A715CT.DSU.utility_slave[0]	PVBusSlave
ARM Cortex-A715CT.MMAP	PVBusLogger
ARM Cortex-A715CT.MMAP.mapper	PVBusMapper
ARM Cortex-A715CT.cpu0	ARM_Cortex-A715
ARM Cortex-A715CT.cpu0.UTLB	TLB
ARM Cortex-A715CT.cpu0.dtlb	TlbCadi
ARM Cortex-A715CT.cpu0.l1dcache	PVCache
ARM Cortex-A715CT.cpu0.l1dcache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexA715CT.cpu0.l1icache	PVCache
ARMCortexA715CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA715CT.cpu0.l2cache	PVCache
ARMCortexA715CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA715CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA715CT.ext_bus	PVBusLogger
ARMCortexA715CT.ext_bus.mapper	PVBusMapper
ARMCortexA715CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-365: ARMCortexA715CT MTI instances

InstanceName	ComponentName
ARMCortexA715CT	ARMv8Cluster
ARMCortexA715CT.AMU	PVBusLogger
ARMCortexA715CT.AMU.mapper	PVBusMapper
ARMCortexA715CT.DAP	PVBusLogger
ARMCortexA715CT.DAP.mapper	PVBusMapper
ARMCortexA715CT.DSU	DSU
ARMCortexA715CT.DSU.PPU_cluster	PPUv1
ARMCortexA715CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA715CT.DSU.PPU_core0	PPUv1
ARMCortexA715CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA715CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA715CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA715CT.DSU.shared_cache	PVCache
ARMCortexA715CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA715CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA715CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA715CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA715CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA715CT.MMAP	PVBusLogger
ARMCortexA715CT.MMAP.mapper	PVBusMapper
ARMCortexA715CT.cpu0	ARM_Cortex-A715
ARMCortexA715CT.cpu0.UTLB	TLB
ARMCortexA715CT.cpu0.l1dcache	PVCache
ARMCortexA715CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA715CT.cpu0.l1icache	PVCache
ARMCortexA715CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA715CT.cpu0.l2cache	PVCache

InstanceName	ComponentName
ARMCortexA715CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA715CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA715CT.ext_bus	PVBusLogger
ARMCortexA715CT.ext_bus.mapper	PVBusMapper
ARMCortexA715CT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder

ARMCortexA715CT contains the following CADI targets:

- ARM_Cortex-A715
- Cluster_ARM_Cortex-A715
- PVCache
- TlbCadi

About ARMCortexA715CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- DynamIQ r3p0.
- L2 cache is supported at the per-core level only and there is no implementation of Core-Complex with shared L2 cache yet.
- P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.
- Utility bus.
- Support for Interrupt signals from SPE is added as `pmbirq[8]`.

Support for the following features is planned for a future release:

- TRBE.
- Transporting architectural metadata tags over TLM2 (OSCI) and AMBA-PV (TLM2 extensions from Arm) bus protocols.
- Core-Complex.
- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.



The `cfgsdisable` signal will be removed in a future release.

The following features will not be implemented:

- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Cache stashing capability.
- Embedded Logic Analyzer (ELA).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Limitations

- Some of the SVE instructions do not raise undefined exceptions when SP is used as a source register.

Ports for ARMCortexA715CT

Table 3-366: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AEND0MP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND1MP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND2MP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND3MP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTART0MP</code>	<code>Value_64</code>	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART1MP</code>	<code>Value_64</code>	Slave	Port to obtain start address of valid peripheral address range (inclusive).

Name	Protocol	Type	Description
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[12]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcsmp_channel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[12]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[12]	Signal	Master	Timer signals to SOC
CNTHVIRQ[12]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[12]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[12]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[12]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[12]	Signal	Master	Timer signals to SOC.
commirq[12]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[12]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmp_channel[12]	PChannel	Master	Core PCSM signals
core_powerdown_out[12]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[12]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[12]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error

Name	Protocol	Type	Description
cryptodisable[12]	Signal	Slave	Disable cryptography extensions after reset.
cti[12]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[12]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[12]	Signal	Master	No power-down request.
dbgpwrupreq[12]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[12]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[12]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[12]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[12]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[12]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[12]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[12]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[12]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[12]	Value_64	Slave	Reset vector base address.

Name	Protocol	Type	Description
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[12]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[12]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[12]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[12]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[12]	Signal	Slave	Virtualised FIQ.
virq[12]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[12]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA715CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance

of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

error_record_feature_register

RAS feature register values. An array of JSON objects. The JSON schema for the array is:

```
[{"ED":0x0, "IMPDEF_3_2":0x0, "UI":0x0, "FI":0x0, "UE":0x0, "CFI":0x0, "CEC":0x0, "RP":0x0, "DUI":0x0, "CEO":0x0, "CI":0x0, "TS":0x0, "INJ":0x0, "FRX":0x0, "UC":0x0, "UEU":0x0, "UER":0x0, "UEO":0x0, "DE":0x0, "CE":0x0, "Visibility":"Core"},other_feature_register_values].
```

Where ED,UI,FI,CE and UE have valid values between 0x0 - 0x3. CFI and

DUI have valid values 0x0, 0x2 and 0x3. CEC has valid values 0x0,0x2 or 0x4.

RP,CEO,INJ,FRX,UC,UEU,UER,UEO,DE has valid values 0x0 or 0x1. CI and TS has valid values of 0x0, 0x1 and 0x2. Visibility has valid values "Core" or "Cluster"

Type

string

Default value

```
[{"ED":0x2,"IMPDEF_3_2":0x1,"UI":0x2,"FI":0x2,"UE":0x1,"CFI":0x2,"CEC":0x2,"RP":0x1,"DUI":0x0,"CEO":0x0,"INJ":0x1,"CI":0x0,"TS":0x0,"FRX":0x0,"UC":0x0,"UEU":0x0,"UER":0x0,"UEO":0x0,"DE":0x0,"CE":0x0,"Visibility":"Cluster"},{"ED":0x2,"UI":0x2,"FI":0x2,"UE":0x1,"CFI":0x2,"CEC":0x2,"RP":0x1,"DUI":0x0,"CEO":0x0,"INJ":0x1,"CI":0x0,"TS":0x0,"FRX":0x0,"UC":0x0,"UEU":0x0,"UER":0x0,"UEO":0x0,"DE":0x0,"CE":0x0,"Visibility":"Cluster"}]
```

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x1

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance

of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

pseudo_fault_generation_feature_register

ARMv8.4 Standard Pseudo-fault generation feature register values. JSON schema for the parameter value is: [{"OF":false, "UC":false, "UEU":false, "UER":false, "UEO":false, "DE":false, "CE":0x0, "CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":false, "NA":false}, other_pseudo-fault_generating_features_register_values]. Where OF, UC, UEU, UER, UEO, DE, CI, ER, PN, AV, MV, SYN, and R have valid false(NOT_SUPPORTED) and true(FEATURE_CONTROLLABLE), where CE can have 0(NOT_SUPPORTED), 1(NONSPECIFIC_CE_SUPPORTED) and 3(TRANSIENT_OR_PERSISTENT_CE_SUPPORTED) and NA can have false(component fakes detection on next access) or true(component fakes detection spontaneously). Effective only when ERXFR's INJ field allows it or has_ras_fault_injection is true.

Type

string

Default value

```
[{"OF":true, "UC":true, "UEU":false, "UER":false, "UEO":false, "DE":0x1,
"CE":0x1,"CI":true, "ER":false, "PN":true, "AV":false, "MV":true, "SYN":true,
"R":true},{ "OF":false, "UC":true, "UEU":false, "UER":false, "UEO":false, "DE":0x1,
"CE":0x1,"CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":true}]
```

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.46 ARM Cortex A720AECT

ARM Cortex A720AECT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-367: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Ports added:

- `cluster_pcs_m_pchannel`
- `core_pcs_m_pchannel`

Parameters added:

- `mpmm_accumulator_multiplier`
- `tlbi_stall_enabled`

Iris and MTI instances for ARMCortexA720AECT

This model has the following Iris instances:

Table 3-368: ARMCortexA720AECT Iris instances

InstanceName	ComponentName
ARMCortexA720AECT	Cluster_ARM_Cortex-A720AE
ARMCortexA720AECT.AMU	PVBUSLogger
ARMCortexA720AECT.AMU.mapper	PVBUSMapper
ARMCortexA720AECT.DAP	PVBUSLogger
ARMCortexA720AECT.DAP.mapper	PVBUSMapper
ARMCortexA720AECT.DSU	DSU
ARMCortexA720AECT.DSU.PPU_cluster	PPUV1
ARMCortexA720AECT.DSU.PPU_cluster.busslave	PVBUSSlave
ARMCortexA720AECT.DSU.PPU_core0	PPUV1
ARMCortexA720AECT.DSU.PPU_core0.busslave	PVBUSSlave
ARMCortexA720AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA720AECT.DSU.mpam_busslave	PVBUSSlave
ARMCortexA720AECT.DSU.shared_cache	PVCACHE
ARMCortexA720AECT.DSU.shared_cache.upstream[0]	PVBUSSlave
ARMCortexA720AECT.DSU.shared_cache.upstream[1]	PVBUSSlave
ARMCortexA720AECT.DSU.shared_cache.upstream[2]	PVBUSSlave
ARMCortexA720AECT.DSU.shared_cache.upstream[3]	PVBUSSlave
ARMCortexA720AECT.DSU.shared_cache.upstream[4]	PVBUSSlave
ARMCortexA720AECT.DSU.utility_slave[0]	PVBUSSlave
ARMCortexA720AECT.MMAP	PVBUSLogger
ARMCortexA720AECT.MMAP.mapper	PVBUSMapper

InstanceName	ComponentName
ARMCortexA720AECT.cpu0	ARM_Cortex-A720AE
ARMCortexA720AECT.cpu0.UTLB	TLB
ARMCortexA720AECT.cpu0.dtlb	TlbCadi
ARMCortexA720AECT.cpu0.l1dcache	PVCache
ARMCortexA720AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA720AECT.cpu0.l1icache	PVCache
ARMCortexA720AECT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA720AECT.cpu0.l2cache	PVCache
ARMCortexA720AECT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA720AECT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA720AECT.ext_bus	PVBusLogger
ARMCortexA720AECT.ext_bus.mapper	PVBusMapper
ARMCortexA720AECT.gic_cpuif_decoder_cluster	GlCv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-369: ARMCortexA720AECT MTI instances

InstanceName	ComponentName
ARMCortexA720AECT	ARMv8Cluster
ARMCortexA720AECT.AMU	PVBusLogger
ARMCortexA720AECT.AMU.mapper	PVBusMapper
ARMCortexA720AECT.DAP	PVBusLogger
ARMCortexA720AECT.DAP.mapper	PVBusMapper
ARMCortexA720AECT.DSU	DSU
ARMCortexA720AECT.DSU.PPU_cluster	PPUv1
ARMCortexA720AECT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA720AECT.DSU.PPU_core0	PPUv1
ARMCortexA720AECT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA720AECT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA720AECT.DSU.mpam_busslave	PVBusSlave
ARMCortexA720AECT.DSU.shared_cache	PVCache
ARMCortexA720AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA720AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA720AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA720AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA720AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA720AECT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA720AECT.MMAP	PVBusLogger
ARMCortexA720AECT.MMAP.mapper	PVBusMapper
ARMCortexA720AECT.cpu0	ARM_Cortex-A720AE

InstanceName	ComponentName
ARMCortexA720AECT.cpu0.UTLB	TLB
ARMCortexA720AECT.cpu0.l1dcache	PVCache
ARMCortexA720AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA720AECT.cpu0.l1icache	PVCache
ARMCortexA720AECT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA720AECT.cpu0.l2cache	PVCache
ARMCortexA720AECT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA720AECT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA720AECT.ext_bus	PVBusLogger
ARMCortexA720AECT.ext_bus.mapper	PVBusMapper
ARMCortexA720AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA720AECT contains the following CADI targets:

- ARM_Cortex-A720AE
- Cluster_ARM_Cortex-A720AE
- PVCache
- TlbCadi

AE-specific features implemented

The only AE-specific features implemented in the model are Split-Lock and Lock-Step, with the following limitations:

- The implementation of Split-Lock and Lock-Step in the model is simply a halving of the number of cores available.
- Lock-step is implemented by a parameter that can be true or false for the entire execution only, rather than by a signal that can be enabled during execution.

As the model does not implement true Split-Lock functionality, the ports and parameters are contiguous. For example, for a 4-core CPU with Split-Lock enabled, `cpu0` and `cpu1` identify the available cores and associated ports, not `cpu0` and `cpu2`.

- Hybrid mode is not modeled in the DSU.

Ports for ARMCortexA720AECT

Table 3-370: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).

Name	Protocol	Type	Description
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[14]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[14]	Signal	Master	Timer signals to SOC
CNTHVIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[14]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[14]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[14]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[14]	Signal	Master	Timer signals to SOC.
commirq[14]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[14]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[14]	PChannel	Master	Core PCSM signals

Name	Protocol	Type	Description
core_powerdown_out[14]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[14]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[14]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[14]	Signal	Slave	Disable cryptography extensions after reset.
cti[14]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[14]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[14]	Signal	Master	No power-down request.
dbgpwrupreq[14]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[14]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[14]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[14]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[14]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[14]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[14]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[14]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.

Name	Protocol	Type	Description
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[14]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[14]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[14]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[14]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[14]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[14]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[14]	Signal	Slave	Virtualised FIQ.
virq[14]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[14]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexA720AECT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

cluster_split_lock_config

Default SPLIT/LOCKED config. The valid values are: 1 - Only LOCKED mode support, 4 - Only SPLIT mode support, 5 - SPLIT or MIXED mode support. Valid only when enable_ae_features is true.

Type

int

Default value

0x1

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_lock_step

Whether the core is configured in Dual Core Lock Step mode (FEAT_DCLS)

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.ETE_REVISION

ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3

Type

int

Default value

0x1

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCSRSTA_FORCED_EXCEP**

TRCSRSTA value for a forcibly traced exception

Type

bool

Default value

0x0

ete.TSMARK

Whether timestamp markers are supported

Type

bool

Default value

0x1

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_impdef_transient_fault_protection

Support the Transient Fault Protection (TFP) flop parity errors through RAS registers (FEAT_TFP)

Type

bool

Default value

0x1

has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

`treat_PAC_as_NOP`

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

`walk_cache_latency`

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.47 **ARMCortexA720CT**

ARMCortexA720CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-371: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Ports added:

- `cluster_pcs_m_pchannel`
- `core_pcs_m_pchannel`

Parameters added:

- `mpmm_accumulator_multiplier`
- `tlbi_stall_enabled`

Iris and MTI instances for ARM Cortex-A720CT

This model has the following Iris instances:

Table 3-372: ARM Cortex-A720CT Iris instances

InstanceName	ComponentName
ARM Cortex-A720CT	Cluster_ARM_Cortex-A720
ARM Cortex-A720CT.AMU	PVBusLogger
ARM Cortex-A720CT.AMU.mapper	PVBusMapper
ARM Cortex-A720CT.DAP	PVBusLogger
ARM Cortex-A720CT.DAP.mapper	PVBusMapper
ARM Cortex-A720CT.DSU	DSU
ARM Cortex-A720CT.DSU.PPU_cluster	PPUv1
ARM Cortex-A720CT.DSU.PPU_cluster.busslave	PVBusSlave
ARM Cortex-A720CT.DSU.PPU_core0	PPUv1
ARM Cortex-A720CT.DSU.PPU_core0.busslave	PVBusSlave
ARM Cortex-A720CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM Cortex-A720CT.DSU.mpam_busslave	PVBusSlave
ARM Cortex-A720CT.DSU.shared_cache	PVCache
ARM Cortex-A720CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM Cortex-A720CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARM Cortex-A720CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARM Cortex-A720CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARM Cortex-A720CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARM Cortex-A720CT.DSU.utility_slave[0]	PVBusSlave
ARM Cortex-A720CT.MMAP	PVBusLogger
ARM Cortex-A720CT.MMAP.mapper	PVBusMapper
ARM Cortex-A720CT.cpu0	ARM_Cortex-A720
ARM Cortex-A720CT.cpu0.UTLB	TLB
ARM Cortex-A720CT.cpu0.dtlb	TlbCadi
ARM Cortex-A720CT.cpu0.l1dcache	PVCache
ARM Cortex-A720CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM Cortex-A720CT.cpu0.l1icache	PVCache
ARM Cortex-A720CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM Cortex-A720CT.cpu0.l2cache	PVCache
ARM Cortex-A720CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARM Cortex-A720CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARM Cortex-A720CT.ext_bus	PVBusLogger
ARM Cortex-A720CT.ext_bus.mapper	PVBusMapper
ARM Cortex-A720CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-373: ARMCortexA720CT MTI instances

InstanceName	ComponentName
ARMCortexA720CT	ARMv8Cluster
ARMCortexA720CT.AMU	PVBusLogger
ARMCortexA720CT.AMU.mapper	PVBusMapper
ARMCortexA720CT.DAP	PVBusLogger
ARMCortexA720CT.DAP.mapper	PVBusMapper
ARMCortexA720CT.DSU	DSU
ARMCortexA720CT.DSU.PPU_cluster	PPUv1
ARMCortexA720CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA720CT.DSU.PPU_core0	PPUv1
ARMCortexA720CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA720CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA720CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA720CT.DSU.shared_cache	PVCache
ARMCortexA720CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA720CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA720CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA720CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA720CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA720CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA720CT.MMAP	PVBusLogger
ARMCortexA720CT.MMAP.mapper	PVBusMapper
ARMCortexA720CT.cpu0	ARM_Cortex-A720
ARMCortexA720CT.cpu0.UTLB	TLB
ARMCortexA720CT.cpu0.l1dcache	PVCache
ARMCortexA720CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA720CT.cpu0.l1icache	PVCache
ARMCortexA720CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA720CT.cpu0.l2cache	PVCache
ARMCortexA720CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA720CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA720CT.ext_bus	PVBusLogger
ARMCortexA720CT.ext_bus.mapper	PVBusMapper
ARMCortexA720CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA720CT contains the following CADI targets:

- ARM_Cortex-A720
- Cluster_ARM_Cortex-A720
- PVCache

- TlbCadi

About ARMCortexA720CT

The model supports the following features:

- L2 cache is supported at the per-core level only and there is no implementation of Core-Complex with shared L2 cache yet.
- Internal PPU support is present.
- A P-Channel for the cluster and for each core.
- BROADCASTPERSIST pin.
- Optional peripheral port.
- Memory-mapped register access to MPAM.
- Per-core clock.
- Utility bus.

Support for the following features is planned for a future release:

- DSU-120 system features are not fully implemented.
- Transporting architectural metadata tags over TLM2 (OSCI) and AMBA-PV (TLM2 extensions from Arm) bus protocols.
- Core-Complex.
- BROADCASTCACHEMAINTPOU pin
- COREINSTRRET and COREINSTRRUN signals.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not supported but signals are implemented.

The following features will not be implemented:

- DynamIQ™ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Cache stashing capability.

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (crypto.dll or crypto.so) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexA720CT

Table 3-374: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[14]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcsmpchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[14]	Signal	Master	Timer signals to SOC
CNTHVIRQ[14]	Signal	Master	Timer signals to SOC.

Name	Protocol	Type	Description
CNTHVSIRQ[14]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[14]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[14]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[14]	Signal	Master	Timer signals to SOC.
commirq[14]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[14]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[14]	PChannel	Master	Core PCSM signals
core_powerdown_out[14]	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the core power info
coreerrirq[14]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[14]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[14]	Signal	Slave	Disable cryptography extensions after reset.
cti[14]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[14]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[14]	Signal	Master	No power-down request.
dbgpwrupreq[14]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[14]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPUs that can be used to reset GICCLK domain components
gicv3_redistributor_s[14]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[14]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[14]	Signal	Master	Interrupt signal from the statistical profiling unit.

Name	Protocol	Type	Description
pmuirq[14]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[14]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[14]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[14]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[14]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[14]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[14]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[14]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[14]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[14]	Signal	Slave	Virtualised FIQ.
virq[14]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[14]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A720CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.ETE_REVISION

ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3

Type

int

Default value

0x1

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCSRSTA_FORCED_EXCEP**

TRCSRSTA value for a forcibly traced exception

Type

bool

Default value

0x0

ete.TSMARK

Whether timestamp markers are supported

Type

bool

Default value

0x1

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use $(n * \text{accumulator value})$ to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_acp

Number of ACP ports

Type

int

Default value

0x0

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.48 ARMCortexA725CT

ARMCortexA725CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-375: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA725CT

This model has the following Iris instances:

Table 3-376: ARMCortexA725CT Iris instances

InstanceName	ComponentName
ARMCortexA725CT	Cluster_ARM_Cortex-A725
ARMCortexA725CT.AMU	PVBusLogger
ARMCortexA725CT.AMU.mapper	PVBusMapper
ARMCortexA725CT.DAP	PVBusLogger
ARMCortexA725CT.DAP.mapper	PVBusMapper
ARMCortexA725CT.DSU	DSU
ARMCortexA725CT.DSU.PPU_cluster	PPUv1
ARMCortexA725CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA725CT.DSU.PPU_core0	PPUv1
ARMCortexA725CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA725CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA725CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA725CT.DSU.shared_cache	PVCache
ARMCortexA725CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA725CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA725CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA725CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA725CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA725CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA725CT.MMAP	PVBusLogger
ARMCortexA725CT.MMAP.mapper	PVBusMapper
ARMCortexA725CT.cpu0	ARM_Cortex-A725
ARMCortexA725CT.cpu0.UTLB	TLB
ARMCortexA725CT.cpu0.dtlb	TlbCadi
ARMCortexA725CT.cpu0.l1dcache	PVCache

InstanceName	ComponentName
ARMCortexA725CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA725CT.cpu0.l1icache	PVCache
ARMCortexA725CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA725CT.cpu0.l2cache	PVCache
ARMCortexA725CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA725CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA725CT.ext_bus	PVBusLogger
ARMCortexA725CT.ext_bus.mapper	PVBusMapper
ARMCortexA725CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-377: ARMCortexA725CT MTI instances

InstanceName	ComponentName
ARMCortexA725CT	ARMv8Cluster
ARMCortexA725CT.AMU	PVBusLogger
ARMCortexA725CT.AMU.mapper	PVBusMapper
ARMCortexA725CT.DAP	PVBusLogger
ARMCortexA725CT.DAP.mapper	PVBusMapper
ARMCortexA725CT.DSU	DSU
ARMCortexA725CT.DSU.PPU_cluster	PPUv1
ARMCortexA725CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA725CT.DSU.PPU_core0	PPUv1
ARMCortexA725CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA725CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA725CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA725CT.DSU.shared_cache	PVCache
ARMCortexA725CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA725CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA725CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA725CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA725CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA725CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA725CT.MMAP	PVBusLogger
ARMCortexA725CT.MMAP.mapper	PVBusMapper
ARMCortexA725CT.cpu0	ARM_Cortex-A725
ARMCortexA725CT.cpu0.UTLB	TLB
ARMCortexA725CT.cpu0.l1dcache	PVCache
ARMCortexA725CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA725CT.cpu0.l1icache	PVCache

InstanceName	ComponentName
ARMCortexA725CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA725CT.cpu0.l2cache	PVCache
ARMCortexA725CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA725CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA725CT.ext_bus	PVBusLogger
ARMCortexA725CT.ext_bus.mapper	PVBusMapper
ARMCortexA725CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexA725CT contains the following CADI targets:

- ARM_Cortex-A725
- Cluster_ARM_Cortex-A725
- PVCache
- TlbCadi

Ports for ARMCortexA725CT

Table 3-378: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[14]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[14]	Signal	Master	Timer signals to SOC
CNTHVIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[14]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[14]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[14]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[14]	Signal	Master	Timer signals to SOC.
commirq[14]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[14]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[14]	PChannel	Master	Core PCSM signals
core_powerdown_out[14]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[14]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[14]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[14]	Signal	Slave	Disable cryptography extensions after reset.
cti[14]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[14]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[14]	Signal	Master	No power-down request.
dbgprupreq[14]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.

Name	Protocol	Type	Description
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[14]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[14]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[14]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[14]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[14]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[14]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[14]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[14]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[14]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[14]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[14]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[14]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[14]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.

Name	Protocol	Type	Description
vfiq[14]	Signal	Slave	Virtualised FIQ.
virq[14]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[14]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM CortexA725CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpu_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-read_latency`

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-size`

L2 Cache size in bytes.

Type

int

Default value

0x100000

`cpuX.l2cache-snoop_data_transfer_latency`

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-snoop_issue_latency`

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.ETE_REVISION

ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3

Type

int

Default value

0x1

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x1

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCSRSTA_FORCED_EXCEP**

TRCSRSTA value for a forcibly traced exception

Type

bool

Default value

0x0

ete.TSMARK

Whether timestamp markers are supported

Type

bool

Default value

0x1

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.49 ARMCortexA725CT_CortexX925CT

ARMCortexA725CT_CortexX925CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-379: IP revisions support

Revision	Quality level
CortexA725 r0p0	Preliminary support
CortexX925 r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexA725CT_CortexX925CT

This model has the following Iris instances:

Table 3-380: ARMCortexA725CT_CortexX925CT Iris instances

InstanceName	ComponentName
ARMCortexA725CT_CortexX925CT	Cluster_ARM_CortexA725_CortexX925_Heterogeneous
ARMCortexA725CT_CortexX925CT.AMU	PVBusLogger
ARMCortexA725CT_CortexX925CT.AMU.mapper	PVBusMapper
ARMCortexA725CT_CortexX925CT.DAP	PVBusLogger
ARMCortexA725CT_CortexX925CT.DAP.mapper	PVBusMapper
ARMCortexA725CT_CortexX925CT.DSU	DSU
ARMCortexA725CT_CortexX925CT.DSU.PPU_cluster	PPUv1
ARMCortexA725CT_CortexX925CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.PPU_core0	PPUv1
ARMCortexA725CT_CortexX925CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.PPU_core1	PPUv1
ARMCortexA725CT_CortexX925CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA725CT_CortexX925CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache	PVCache
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.MMAP	PVBusLogger
ARMCortexA725CT_CortexX925CT.MMAP.mapper	PVBusMapper
ARMCortexA725CT_CortexX925CT.ext_bus	PVBusLogger
ARMCortexA725CT_CortexX925CT.ext_bus.mapper	PVBusMapper
ARMCortexA725CT_CortexX925CT.gic_cpuif_decoder_cluster	GCv3CPUInterfaceDecoder
ARMCortexA725CT_CortexX925CT.subcluster0	Subcluster_ARM_Cortex-A725

InstanceName	ComponentName
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0	ARM_Cortex-A725
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.dtlb	TlbCadi
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster1	Subcluster_ARM_Cortex-X925
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0	ARM_Cortex-X925
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster1.cpu1.dtlb	TlbCadi

This model has the following MTI trace components:

Table 3-381: ARMCortexA725CT_CortexX925CT MTI instances

InstanceName	ComponentName
ARMCortexA725CT_CortexX925CT	ARMv8Cluster
ARMCortexA725CT_CortexX925CT.AMU	PVBusLogger
ARMCortexA725CT_CortexX925CT.AMU.mapper	PVBusMapper
ARMCortexA725CT_CortexX925CT.DAP	PVBusLogger
ARMCortexA725CT_CortexX925CT.DAP.mapper	PVBusMapper
ARMCortexA725CT_CortexX925CT.DSU	DSU
ARMCortexA725CT_CortexX925CT.DSU.PPU_cluster	PPUv1
ARMCortexA725CT_CortexX925CT.DSU.PPU_cluster.busslave	PVBusSlave

InstanceName	ComponentName
ARMCortexA725CT_CortexX925CT.DSU.PPU_core0	PPUv1
ARMCortexA725CT_CortexX925CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.PPU_core1	PPUv1
ARMCortexA725CT_CortexX925CT.DSU.PPU_core1.busslave	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexA725CT_CortexX925CT.DSU.mpam_busslave	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache	PVCache
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[5]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.shared_cache.upstream[6]	PVBusSlave
ARMCortexA725CT_CortexX925CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.MMAP	PVBusLogger
ARMCortexA725CT_CortexX925CT.MMAP.mapper	PVBusMapper
ARMCortexA725CT_CortexX925CT.ext_bus	PVBusLogger
ARMCortexA725CT_CortexX925CT.ext_bus.mapper	PVBusMapper
ARMCortexA725CT_CortexX925CT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0	ARM_Cortex-A725
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.UTLB	TLB
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l1dcache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l1icache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster0.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0	ARM_Cortex-X925
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.UTLB	TLB
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l1dcache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l1icache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache	PVCache
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexA725CT_CortexX925CT.subcluster1.cpu0.l2cache.upstream[1]	PVBusSlave

ARMCortexA725CT_CortexX925CT contains the following CADI targets:

- ARM_Cortex-A725
- ARM_Cortex-X925
- Cluster_ARM_CortexA725_CortexX925_Heterogeneous
- PVCache
- Subcluster_ARM_Cortex-A725
- Subcluster_ARM_Cortex-X925
- TlbCadi

Ports for ARMCortexA725CT_CortexX925CT

Table 3-382: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[26]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcsmpchannel	PChannel	Master	Cluster PCSM signal

Name	Protocol	Type	Description
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[26]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[26]	Signal	Master	Timer signals to SOC
CNTHVIRQ[26]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[26]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[26]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[26]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[26]	Signal	Master	Timer signals to SOC.
commirq[26]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[26]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[26]	PChannel	Master	Core PCSM signals
core_powerdown_out[26]	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the core power info
coreerrirq[26]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[26]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[26]	Signal	Slave	Disable cryptography extensions after reset.
cti[26]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[26]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[26]	Signal	Master	No power-down request.
dbgpwrupreq[26]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[26]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.

Name	Protocol	Type	Description
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[26]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[26]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[26]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[26]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[26]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[26]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[26]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[26]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[26]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[26]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[26]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[26]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[26]	Signal	Slave	Virtualised FIQ.
virq[26]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).

Name	Protocol	Type	Description
vsei[26]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM Cortex A725CT_Cortex X925CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use $(n * \text{accumulator value})$ to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

subcluster0.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster0.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster0.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster0.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster0.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster0.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster0.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster0.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster0.cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

subcluster0.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x100000

subcluster0.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster0.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster0.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster0.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster0.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster0.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster0.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster0.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster0.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster0.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster0.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster0.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster0.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster0.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster0.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster0.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster0.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster0.ete.ETE_REVISION

ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3

Type

int

Default value

0x1

subcluster0.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster0.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster0.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

subcluster0.ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

subcluster0.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster0.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster0.ete.RETSTACK

Return stack depth

Type

int

Default value

0x1

subcluster0.ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

subcluster0.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster0.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster0.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster0.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster0.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster0.ete.TSMARK

Whether timestamp markers are supported

Type

bool

Default value

0x1

subcluster0.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster0.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster0.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster0.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster0.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster0.has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster0.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

subcluster0.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster0.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster0.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster0.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster0.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster0.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

subcluster0.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster0.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster0.tlbi_stall_enabled

If true, tlbi invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster0.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Branch, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster0.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

subcluster1.NUM_CORES

Number of cores in the subcluster. Total number of cores in cluster may not exceed 14.

Type

int

Default value

0x1

subcluster1.cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

subcluster1.cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

subcluster1.cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

subcluster1.cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

subcluster1.cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

subcluster1.cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

subcluster1.cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

subcluster1.cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

subcluster1.cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

subcluster1.cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

subcluster1.cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

subcluster1.cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

subcluster1.cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

subcluster1.cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**subcluster1.cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**subcluster1.cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

subcluster1.cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

subcluster1.cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

subcluster1.cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

subcluster1.cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

subcluster1.cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

subcluster1.cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

subcluster1.cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

subcluster1.dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

subcluster1.dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

subcluster1.dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.ecv_support_level

Implement Enhanced Counter Virtualization feature from ARMv8.6. 0, Not supported. 1, fully supported without CNTPOFF. 2, fully supported with CNTPOFF (FEAT_ECV).

Type

int

Default value

0x2

subcluster1.ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

subcluster1.ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

subcluster1.ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

subcluster1.ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

subcluster1.ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

subcluster1.ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

subcluster1.ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

subcluster1.ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

subcluster1.ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

subcluster1.ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

subcluster1.ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**subcluster1.ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

subcluster1.ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x2

subcluster1.force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

subcluster1.force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

subcluster1.force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

subcluster1.has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

subcluster1.has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

subcluster1.has_mt_pmu_disable_feature

Implements Multi-threading PMU disable extension from ARMv8.6 (FEAT_MTPMU). 0: FEAT_MTPMU is disabled, 1: FEAT_MTPMU is enabled if ARMv8.6 is implemented, 2: FEAT_MTPMU is cherry-picked, 0xF: The feature is disabled and is represented by value 0xF in ID_AA64DFR0_EL1.MTPMU.

Type

int

Default value

0x0

subcluster1.has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

subcluster1.has_v8_7_spe_inverted_filtering

Where FEAT_SPEv1p2 is implemented, whether the inverting filtering by events

Type

bool

Default value

0x0

subcluster1.has_v8_7_spe_previous_branch_target

Where FEAT_SPEv1p2 is implemented, whether the optional branch target feature is implemented

Type

bool

Default value

0x0

subcluster1.icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

subcluster1.icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

subcluster1.icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

subcluster1.icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

subcluster1.instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

subcluster1.memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

subcluster1.mpam_has_altsp

MPAM Whether MPAMIDR_EL1.HAS_ALTSP bit is set or clear

Type

bool

Default value

0x0

subcluster1.mpamidr_has_force_ns

Whether MPAMIDR_EL1.HAS_FORCE_NS bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

subcluster1.mpamidr_has_sdeflt

Whether MPAMIDR_EL1.HAS_SDEFLT bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

subcluster1.mpamidr_has_tidr

Whether MPAMIDR_EL1.HAS_TIDR bit is set or clear

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

subcluster1.pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x1f

subcluster1.ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

subcluster1.tcr_txsz_undersize_should_fault

If large VA is not supported, Whether undersized TxSZ value should generate translation fault

Type

bool

Default value

0x0

subcluster1.tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

subcluster1.tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

subcluster1.treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

subcluster1.walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.50 ARM CortexM0CT

ARM CortexM0CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-383: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARM CortexM0CT

This model has the following Iris instances:

Table 3-384: ARM CortexM0CT Iris instances

InstanceName	ComponentName
ARM CortexM0CT	ARM_Cortex-M0
ARM CortexM0CT.acp_mapper	PVBusMapper
ARM CortexM0CT.ext_bus	PVBusLogger
ARM CortexM0CT.ext_bus.mapper	PVBusMapper
ARM CortexM0CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-385: ARM CortexM0CT MTI instances

InstanceName	ComponentName
ARM CortexM0CT	ARM_Cortex-M0
ARM CortexM0CT.acp_mapper	PVBusMapper
ARM CortexM0CT.ext_bus	PVBusLogger
ARM CortexM0CT.ext_bus.mapper	PVBusMapper
ARM CortexM0CT.l2_flusher	AsyncCacheFlushUnit

ARM CortexM0CT contains the following CADI targets:

- [ARM_Cortex-M0](#)

Differences between the model and the RTL

- This model does not have a parameter that is equivalent to the RAR integration option. The architecturally-required register state is reset.
- This model exposes a VTOR register through CADI but this register does not exist in the IP.

- Arm®v6-M is a subset of Armv7-M. Arm does not guarantee that all Armv7-M-specific behavior is absent from Armv6-M Fast Models cores. Therefore, Arm does not guarantee that code that runs on Armv7-M cores but fails on Armv6-M cores will also fail on Armv6-M Fast Models cores.

Ports for ARM Cortex-M0CT

Table 3-386: Ports

Name	Protocol	Type	Description
ahbd	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
currpri	Value	Master	Current execution priority.
edbgrq	Signal	Slave	External debug request.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
intisr[32]	Signal	Slave	This signal array delivers signals to the NVIC.
intnmi	Signal	Slave	Configure non maskable interrupt.
lockup	Signal	Master	Asserted when the processor is in lockup state.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbus_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
stcalib	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.

Parameters for ARM Cortex-M0CT

BIGENDINIT

Initialize processor to big endian mode

Type

bool

Default value

0x0

BKPT

Number of breakpoint unit comparators implemented

Type

int

Default value

0x4

DBG

Set whether debug extensions are implemented

Type

bool

Default value

0x1

NUM_IRQ

Number of user interrupts

Type

int

Default value

0x20

SYST

Enable support for SysTick timer functionality

Type

bool

Default value

0x1

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

WPT

Number of watchpoint unit comparators implemented

Type

int

Default value

0x2

cpi_div

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value
0x20700000

semihosting-stack_base
Virtual address of base of descending stack

Type
int

Default value
0x20800000

semihosting-stack_limit
Virtual address of stack limit

Type
int

Default value
0x20700000

3.5.51 ARMCortexM0PlusCT

ARMCortexM0PlusCT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-387: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexM0PlusCT

This model has the following Iris instances:

Table 3-388: ARMCortexM0PlusCT Iris instances

InstanceName	ComponentName
ARMCortexM0PlusCT	ARM_Cortex-M0+
ARMCortexM0PlusCT.acp_mapper	PVBusMapper
ARMCortexM0PlusCT.ext_bus	PVBusLogger
ARMCortexM0PlusCT.ext_bus.mapper	PVBusMapper
ARMCortexM0PlusCT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-389: ARMCortexM0PlusCT MTI instances

InstanceName	ComponentName
ARMCortexM0PlusCT	ARM_Cortex-M0plus
ARMCortexM0PlusCT.acp_mapper	PVBusMapper
ARMCortexM0PlusCT.ext_bus	PVBusLogger
ARMCortexM0PlusCT.ext_bus.mapper	PVBusMapper
ARMCortexM0PlusCT.l2_flusher	AsyncCacheFlushUnit

ARMCortexM0PlusCT contains the following CADI targets:

- ARM_Cortex-M0+

Differences between the model and the RTL

- This model does not have a parameter that is equivalent to the RAR integration option. The architecturally required register state is reset.
- This model does not have a parameter that is equivalent to the RAR integration option. The architecturally-required register state is reset.
- This model exposes a VTOR register through CADI but this register does not exist in the IP.
- Arm®v6-M is a subset of Armv7-M. Arm does not guarantee that all Armv7-M-specific behavior is absent from Armv6-M Fast Models cores. Therefore, Arm does not guarantee that code that runs on Armv7-M cores but fails on Armv6-M cores will also fail on Armv6-M Fast Models cores.

Ports for ARMCortexM0PlusCT

Table 3-390: Ports

Name	Protocol	Type	Description
ahbd	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
cpuwait	Signal	Slave	CPUWAIT extends effect of reset when true
currpri	Value	Master	Current execution priority.
edbgrq	Signal	Slave	External debug request.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
intisr[32]	Signal	Slave	This signal array delivers signals to the NVIC.
intnmi	Signal	Slave	Configure non maskable interrupt.
io_port_in	PVBus	Slave	I/O port pair. See the documentation for the io_port_out port.

Name	Protocol	Type	Description
io_port_out	PVBus	Master	I/O port pair. Used if IOP is true. Transactions from io_port_out which do not "match" should be returned via io_port_in. For performance reasons, the I/O port interface is not modelled directly. Instead, a simple PVBus gasket is inserted at the point in the memory system where the I/O port would be. In hardware, a device would be attached to the port which would tell the CPU whether it would like to intercept each transaction, given its address. This can be modelled in a performant manner by connecting a PVBusMapper-based device to io_port_out which intercepts transactions of interest and passes other transactions back to the CPU via io_port_in. Your I/O port device model is also responsible for aborting transactions which would be aborted on hardware (e.g. exclusives) if necessary.
lockup	Signal	Master	Asserted when the processor is in lockup state.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbuse_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
stcalib	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.

Parameters for ARMCortexM0PlusCT

BIGENDINIT

Initialize processor to big endian mode

Type

bool

Default value

0x0

BKPT

Number of breakpoint unit comparators implemented

Type

int

Default value

0x4

DBG

Set whether debug extensions are implemented

Type

bool

Default value

0x1

IOP

Send all d-side transactions to the port, io_port_out. Transactions which do not match should be returned to the port, io_port_in

Type

bool

Default value

0x0

IRQDIS

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n]

Type

int

Default value

0x0

NUM_IRQ

Number of user interrupts

Type

int

Default value

0x20

NUM_MPU_REGION

Number of MPU regions

Type

int

Default value

0x0

SYST

Enable support for SysTick timer functionality

Type

bool

Default value

0x1

USER

Enable support for Unprivileged/Privileged Extension

Type

bool

Default value

0x0

VTOR

Include Vector Table Offset Register

Type

bool

Default value

0x0

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

WPT

Number of watchpoint unit comparators implemented

Type

int

Default value

0x2

cpi_div

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

3.5.52 ARMCortexM3CT

ARMCortexM3CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-391: IP revisions support

Revision	Quality level
r2p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexM3CT

This model has the following Iris instances:

Table 3-392: ARMCortexM3CT Iris instances

InstanceName	ComponentName
ARMCortexM3CT	ARM_Cortex-M3
ARMCortexM3CT.acp_mapper	PVBusMapper
ARMCortexM3CT.ext_bus	PVBusLogger
ARMCortexM3CT.ext_bus.mapper	PVBusMapper
ARMCortexM3CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-393: ARMCortexM3CT MTI instances

InstanceName	ComponentName
ARMCortexM3CT	ARM_Cortex-M3
ARMCortexM3CT.acp_mapper	PVBusMapper
ARMCortexM3CT.ext_bus	PVBusLogger
ARMCortexM3CT.ext_bus.mapper	PVBusMapper
ARMCortexM3CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexM3CT contains the following CADI targets:

- ARM_Cortex-M3

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- The WIC is not currently implemented.
- Power control is not implemented, so the processor does not set the SLEEPING or SLEEPDEEP signals. It does not support powering down of the processor.
- Only the minimal level of debug support is provided (no DAP, FPB, DWT, or halting debug capability).
- Debug-related components are not implemented.
- The unimplemented registers are the processor debug registers, system debug registers, debug interface port registers, TPIU registers, and ETM registers.
- The processor must still be clocked even if it has asserted the sleeping or sleepdeep signals.
- Disabling processor features using the Auxiliary Control Register is not supported.
- Only a single `pvbus_m` master port is provided. This combines the ICode, DCode, and System bus interfaces of the RTL. The external PPB bus is provided by the `pv_ppbus_m` master port.
- In privileged mode, STRT and LDRT to the PPB region are not forbidden access.
- No support for ETM, TPIU, or HTM.
- There is no supported equivalent of the RESET_ALL_REGS configuration setting in RTL (that forces all registers to have a well-defined value on reset).
- The RTL implements the ROM table as an external component on the External Private Peripheral Bus. In the CT model, the ROM table is implemented internally as a fallback if an external PPB access in the ROM table address region aborts. This permits the default ROM table to be overridden (by implementing an external component connected to the external PPB to handle accesses to these addresses) without requiring every user of the processor to implement and connect a ROM table component.

Implementation of ITM in M-class models

This model has a parameter that enables partial support for Instrumentation Trace Macrocell (ITM). In hardware or RTL, trace data from ITM is sent in packets to the trace block serially using a single pin or wire. In the model, if it is enabled, the ITM trace data is output using an MTI trace source called `itm`. The `itm` trace source has an `itm_packet_type` field. The following table shows which packet types the model supports:

Table 3-394: ITM_PACKET_TYPE field values that the model supports

Field value	Description	Supported by model
ITM_SYNC	Synchronization packet	Not supported.
ITM_P_OVERFLOW	Protocol: Overflow packet	Not supported.
ITM_P_LOCAL_TIMESTAMP	Protocol: Local timestamp packets	Not supported.
ITM_P_GLOBAL_TIMESTAMP	Protocol: Global timestamp packets	Not supported.
ITM_P_EXTEN	Protocol: Extension packet	Not supported.
ITM_S_INSTRUMENTATION	Source: Instrumentation packet	Supported.

Field value	Description	Supported by model
ITM_S_DWT_EVENT_COUNTER	Hardware source: Event counter wrapping	Not supported.
ITM_S_DWT_EXCEPTION	Hardware source: Exception tracing	Supported.
ITM_S_DWT_PC_SAMPLING	Hardware source: PC sampling	Not supported.
ITM_S_DWT_DATA_PC_TRACE	Hardware source: DWT Data trace PC value	Supported.
ITM_S_DWT_DATA_ADDRESS_TRACE	Hardware source: DWT Data trace address value	Supported.
ITM_S_DWT_DATA_DATA_TRACE	Hardware source: DWT Data trace DATA value	Supported.

Ports for ARMCortexM3CT

Table 3-395: Ports

Name	Protocol	Type	Description
ahb_ap	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
auxfault	Value	Slave	This is wired to the Auxiliary Fault Status Register.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
currpri	Value	Master	Current execution priority.
edbgrq	Signal	Slave	External debug request.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
intisr[240]	Signal	Slave	This signal array delivers signals to the NVIC.
intnmi	Signal	Slave	Configure non maskable interrupt.
lockup	Signal	Master	Asserted when the processor is in lockup state.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbus_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
stcalib	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.

Parameters for ARMCortexM3CT

BB_PRESENT

Enable bitbanding

Type

bool

Default value

0x1

BIGENDINIT

Initialize processor to big endian mode

Type

bool

Default value

0x0

DBG_LVL

0: No debug; 1: Minimal debug; 2: Full debug without DWT address matching; 3: Full debug support with DWT data-comparators

Type

int

Default value

0x3

LVL_WIDTH

Number of bits of interrupt priority

Type

int

Default value

0x3

NUM_IRQ

Number of user interrupts

Type

int

Default value

0x10

NUM_MPU_REGION

Number of MPU regions

Type

int

Default value

0x8

TRACE_LVL

Level of instrumentation trace supported. 0: No trace, 1: Standard trace. ITM, TPIU and DWT triggers and counters present, 2: Full trace. ITM, TPIU, ETM and DWT triggers and counters present

Type

int

Default value

0x1

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

cpi_div

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CPUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CPUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

3.5.53 ARMCortexM4CT

ARMCortexM4CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-396: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexM4CT

This model has the following Iris instances:

Table 3-397: ARMCortexM4CT Iris instances

InstanceName	ComponentName
ARMCortexM4CT	ARM_Cortex-M4
ARMCortexM4CT.acp_mapper	PVBusMapper
ARMCortexM4CT.ext_bus	PVBusLogger
ARMCortexM4CT.ext_bus.mapper	PVBusMapper
ARMCortexM4CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-398: ARMCortexM4CT MTI instances

InstanceName	ComponentName
ARMCortexM4CT	ARM_Cortex-M4
ARMCortexM4CT.acp_mapper	PVBusMapper
ARMCortexM4CT.ext_bus	PVBusLogger
ARMCortexM4CT.ext_bus.mapper	PVBusMapper
ARMCortexM4CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexM4CT contains the following CADI targets:

- ARM_Cortex-M4

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- The Wakeup Interrupt Controller (WIC) is not implemented.
- Power control is not implemented. Powering down of the processor is not supported. The processor must still be clocked even if it has asserted the sleeping or sleepdeep signals.
- Only the minimal level of debug support is provided (no DAP, FPB, DWT, or halting debug capability).
- No debug-related components are implemented.
- The unimplemented registers are the processor debug registers, system debug registers, debug interface port registers, TPIU registers, and ETM registers.
- No support for ETM, TPIU, or HTM.

- There is no supported equivalent of the RESET_ALL_REGS configuration setting in RTL (that forces all registers to have a well-defined value on reset).
- Disabling processor features using the Auxiliary Control Register is not supported.
- Only a single `p_vbus_m` master port is provided. This combines the ICode, DCode, and System bus interfaces of the RTL. The external PPB bus is provided by the `p_v_ppbus_m` master port.
- In privileged mode, STRT and LDRT to the PPB region are not forbidden access.
- The RTL implements the ROM table as an external component on the External Private Peripheral Bus. In the CT model, the ROM table is implemented internally as a fallback if an external PPB access in the ROM table address region aborts. This permits the default ROM table to be overridden (by implementing an external component connected to the external PPB to handle accesses to these addresses) without requiring every user of the processor to implement and connect a ROM table component.
- Because the CT model does not provide a DAP port or halting debug capability, the `dbggen` signal is ignored.

Implementation of ITM in M-class models

This model has a parameter that enables partial support for Instrumentation Trace Macrocell (ITM). In hardware or RTL, trace data from ITM is sent in packets to the trace block serially using a single pin or wire. In the model, if it is enabled, the ITM trace data is output using an MTI trace source called `itm`. The `itm` trace source has an `ITM_PACKET_TYPE` field. The following table shows which packet types the model supports:

Table 3-399: ITM_PACKET_TYPE field values that the model supports

Field value	Description	Supported by model
ITM_SYNC	Synchronization packet	Not supported.
ITM_P_OVERFLOW	Protocol: Overflow packet	Not supported.
ITM_P_LOCAL_TIMESTAMP	Protocol: Local timestamp packets	Not supported.
ITM_P_GLOBAL_TIMESTAMP	Protocol: Global timestamp packets	Not supported.
ITM_P_EXTEN	Protocol: Extension packet	Not supported.
ITM_S_INSTRUMENTATION	Source: Instrumentation packet	Supported.
ITM_S_DWT_EVENT_COUNTER	Hardware source: Event counter wrapping	Not supported.
ITM_S_DWT_EXCEPTION	Hardware source: Exception tracing	Supported.
ITM_S_DWT_PC_SAMPLING	Hardware source: PC sampling	Not supported.
ITM_S_DWT_DATA_PC_TRACE	Hardware source: DWT Data trace PC value	Supported.
ITM_S_DWT_DATA_ADDRESS_TRACE	Hardware source: DWT Data trace address value	Supported.
ITM_S_DWT_DATA_DATA_TRACE	Hardware source: DWT Data trace DATA value	Supported.

Ports for ARMCortexM4CT

Table 3-400: Ports

Name	Protocol	Type	Description
ahb_ap	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
auxfault	Value	Slave	This is wired to the Auxiliary Fault Status Register.

Name	Protocol	Type	Description
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
currpri	Value	Master	Current execution priority.
dbgen	Signal	Slave	Disallow (DAP) debugger access.
edbgrq	Signal	Slave	External debug request.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
fpudisable	Signal	Slave	Configure core with no FPU on reset.
fpxxc	Value	Master	Port that sends the value of the FPxxC exception flags (FPIXC, FPIDC, FPOFC, FPUFC, FPDZC, FPIOC). Each flag is present in the same position as in the FPSCR register (e.g. fpxxc[0] = FPIOC).
intisr[240]	Signal	Slave	This signal array delivers signals to the NVIC.
intnmi	Signal	Slave	Configure non maskable interrupt.
lockup	Signal	Master	Asserted when the processor is in lockup state.
mpudisable	Signal	Slave	Configure core with no MPU on reset.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbus_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
stcalib	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.

Parameters for ARM CortexM4CT

BB_PRESENT

Enable bitbanding

Type

bool

Default value

0x1

BIGENDINIT

Initialize processor to big endian mode

Type

bool

Default value

0x0

DBG_LVL

0: No debug; 1: Minimal debug; 2: Full debug without DWT address matching; 3: Full debug support with, DWT can compare data as well as address

Type

int

Default value

0x3

LVL_WIDTH

Number of bits of interrupt priority

Type

int

Default value

0x3

NUM_IRQ

Number of user interrupts

Type

int

Default value

0x10

NUM_MPU_REGION

Number of MPU regions

Type

int

Default value

0x8

TRACE_LVL

Level of instrumentation trace supported. 0: No trace, 1: Standard trace. ITM, TPIU and DWT triggers and counters present, 2: Full trace. ITM, TPIU, ETM and DWT triggers and counters present

Type

int

Default value

0x1

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

cpi_div

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value`0xffffffffffffffff`**reported_revision_number**

Purely cosmetic revision number value to be displayed from the CPUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type`int`**Default value**`0xffffffffffffffff`**semihosting-Thumb_SVC**

T32 SVC number for semihosting

Type`int`**Default value**`0xab`**semihosting-cmd_line**

Command line available to semihosting SVC calls

Type`string`**Default value****semihosting-cwd**

Base directory for semihosting file access.

Type`string`**Default value****semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type`bool`**Default value**`0x1`**semihosting-heap_base**

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

3.5.54 ARMCortexM7CT

ARMCortexM7CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-401: IP revisions support

Revision	Quality level
r1p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexM7CT

This model has the following Iris instances:

Table 3-402: ARMCortexM7CT Iris instances

InstanceName	ComponentName
ARMCortexM7CT	ARM_Cortex-M7
ARMCortexM7CT.acp_mapper	PVBusMapper
ARMCortexM7CT.ext_bus	PVBusLogger
ARMCortexM7CT.ext_bus.mapper	PVBusMapper
ARMCortexM7CT.l1_incoherent_interconnect	PVCache
ARMCortexM7CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexM7CT.l1dcache	PVCache
ARMCortexM7CT.l1dcache.upstream[0]	PVBusSlave
ARMCortexM7CT.l1icache	PVCache
ARMCortexM7CT.l1icache.upstream[0]	PVBusSlave
ARMCortexM7CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-403: ARMCortexM7CT MTI instances

InstanceName	ComponentName
ARMCortexM7CT	ARM_Cortex-M7
ARMCortexM7CT.acp_mapper	PVBusMapper
ARMCortexM7CT.ext_bus	PVBusLogger
ARMCortexM7CT.ext_bus.mapper	PVBusMapper
ARMCortexM7CT.l1_incoherent_interconnect	PVCache
ARMCortexM7CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexM7CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexM7CT.l1dcache	PVCache
ARMCortexM7CT.l1dcache.upstream[0]	PVBusSlave
ARMCortexM7CT.l1icache	PVCache
ARMCortexM7CT.l1icache.upstream[0]	PVBusSlave
ARMCortexM7CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexM7CT contains the following CADI targets:

- ARM_Cortex-M7

Differences between the model and the RTL

- TCMs are modeled internally and the model does not support external TCMs or the ports associated with them.
- ECC support is hardware-specific so is not modeled.

Implementation of ITM in M-class models

This model has a parameter that enables partial support for Instrumentation Trace Macrocell (ITM). In hardware or RTL, trace data from ITM is sent in packets to the trace block serially using a single pin or wire. In the model, if it is enabled, the ITM trace data is output using an MTI trace source called `itm`. The `itm` trace source has an `itm_packet_type` field. The following table shows which packet types the model supports:

Table 3-404: ITM_PACKET_TYPE field values that the model supports

Field value	Description	Supported by model
ITM_SYNC	Synchronization packet	Not supported.
ITM_P_OVERFLOW	Protocol: Overflow packet	Not supported.
ITM_P_LOCAL_TIMESTAMP	Protocol: Local timestamp packets	Not supported.
ITM_P_GLOBAL_TIMESTAMP	Protocol: Global timestamp packets	Not supported.
ITM_P_EXTEN	Protocol: Extension packet	Not supported.
ITM_S_INSTRUMENTATION	Source: Instrumentation packet	Supported.
ITM_S_DWT_EVENT_COUNTER	Hardware source: Event counter wrapping	Not supported.
ITM_S_DWT_EXCEPTION	Hardware source: Exception tracing	Supported.
ITM_S_DWT_PC_SAMPLING	Hardware source: PC sampling	Not supported.
ITM_S_DWT_DATA_PC_TRACE	Hardware source: DWT Data trace PC value	Supported.
ITM_S_DWT_DATA_ADDRESS_TRACE	Hardware source: DWT Data trace address value	Supported.
ITM_S_DWT_DATA_DATA_TRACE	Hardware source: DWT Data trace DATA value	Supported.

Ports for ARMCortexM7CT

Table 3-405: Ports

Name	Protocol	Type	Description
ahbd	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
ahbp_m	PVBus	Master	The core will generate Vendor System data accesses on this port.
ahbs	PVBus	Slave	External master (e.g. DMA) can write TCMs (whether or not enabled in xTCMCR).
auxfault	Value	Slave	This is wired to the Auxiliary Fault Status Register.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
cpuwait	Signal	Slave	When this signal is HIGH out of reset, it forces the processor into a quiescent state that delays its boot-up sequence and instruction execution until this signal is driven LOW
currpri	Value	Master	Current execution priority.
dap_s	PVBus	Slave	Debug Access Port (DAP).
dbgen	Signal	Slave	Invasive debug enable.
dbgrestart	Signal	Slave	External debug request.
dbgrestarted	Signal	Master	External debug request.
edbgrq	Signal	Slave	External debug request.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.

Name	Protocol	Type	Description
fpudisable	Signal	Slave	Configure core with no FPU on reset.
fpxxc	Value	Master	Port that sends the value of the FPxxC exception flags (FPIX, FPIDC, FPOFC, FPUFC, FPDZC, FPIOC). Each flag is present in the same position as in the FPSCR register (e.g. fpxxc[0] = FPIOC).
halted	Signal	Master	External debug request.
initahbpen	Signal	Slave	Enable AHBP on the next reset
initvtor	Value	Slave	Initial value of the Vector Table Offset Register (VTOR)
intisr[240]	Signal	Slave	This signal array delivers signals to the NVIC.
intnmi	Signal	Slave	Configure non maskable interrupt.
lockup	Signal	Master	Asserted when the processor is in lockup state.
mpudisable	Signal	Slave	Configure core with no MPU on reset.
niden	Signal	Slave	Non-invasive debug enable.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pdbus_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
stcalib	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.

Parameters for ARM CortexM7CT

BIGENDINIT

Initialize processor to big endian mode

Type

bool

Default value

0x0

CFGAHBPSZ

Size of the AHBP port memory region. 0=AHBP disabled, 1=64MB, 2=128MB, 3=256MB, 4=512MB

Type

int

Default value

0x0

CTI

CTI (Cross Trigger Interface) included

Type

bool

Default value

0x0

CTI_irq0_pin

CTI interrupt request 0 pin

Type

int

Default value

0x4

CTI_irq1_pin

CTI interrupt request 1 pin

Type

int

Default value

0x5

DBGLVL

0: 2 DWT, 4 FPB; 1: 4 DWT, 8 FPB comparators

Type

int

Default value

0x1

DP_FLOAT

Support 8-byte floats

Type

bool

Default value

0x1

INITAHBPEN

The AHBP enable state at reset

Type

bool

Default value

0x0

INITVTOR

vector-table offset at reset

Type

int

Default value

0x0

LVL_WIDTH

Number of bits of interrupt priority

Type

int

Default value

0x3

NUM_IRQ

Number of user interrupts

Type

int

Default value

0x20

NUM_MPU_REGION

Number of MPU regions

Type

int

Default value

0x10

TRC

Level of instrumentation trace supported. false : No ITM trace included, true: ITM trace included

Type

bool

Default value

0x1

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

cpi_div

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

dcache-size

L1 D-cache size in bytes

Type

int

Default value

0x8000

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dtcm_enable

Enable DTCM at reset

Type

bool

Default value

0x0

dtcm_size

DTCM size in KB

Type

int

Default value

0x100

icache-size

L1 I-cache size in bytes

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

itcm_enable

Enable ITCM at reset

Type

bool

Default value

0x0

itcm_size

ITCM size in KB

Type

int

Default value

0x100

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

3.5.55 ARMCortexM23CT

ARMCortexM23CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-406: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexM23CT

This model has the following Iris instances:

Table 3-407: ARMCortexM23CT Iris instances

InstanceName	ComponentName
ARMCortexM23CT	ARM_Cortex-M23
ARMCortexM23CT.acp_mapper	PVBusMapper
ARMCortexM23CT.ext_bus	PVBusLogger
ARMCortexM23CT.ext_bus.mapper	PVBusMapper
ARMCortexM23CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-408: ARMCortexM23CT MTI instances

InstanceName	ComponentName
ARMCortexM23CT	ARM_Cortex-M23
ARMCortexM23CT.acp_mapper	PVBusMapper
ARMCortexM23CT.ext_bus	PVBusLogger
ARMCortexM23CT.ext_bus.mapper	PVBusMapper
ARMCortexM23CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexM23CT contains the following CADI targets:

- ARM_Cortex-M23

Differences between the CT model and RTL implementations

The model does not support MTB, ETM, or TPIU. MTB RAM is absent on the model.

Ports for ARMCortexM23CT

Table 3-409: Ports

Name	Protocol	Type	Description
ahbd	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
core_dside_bus_gasket_in	PVBus	Slave	-
core_dside_bus_gasket_out	PVBus	Master	-
cpuwait	Signal	Slave	Clear = Core goes through reset sequence as normal, Set = Core waits out of reset.
currpri	Value	Master	Current execution priority.
dap_s	PVBus	Slave	Debug Access Port (DAP).
dbgen	Signal	Slave	Invasive debug control signals. Debug enable, Set=enabled, Clear=disabled
dbgrestart	Signal	Slave	External request to leave debug state
dbgrestarted	Signal	Master	Acknowledge for DBGRESTART
edbgrq	Signal	Slave	External debug request.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
halted	Signal	Master	Core is in halt mode debug state
hreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
idau	PVBus	Master	The core will generate IDAU requests on this port.
idau_invalidate_region	Value_64	Slave	64 bit number to invalid IDAU memory ranage (start_address<<32 end_address)
initvtor	Value	Slave	Reset configuration port - Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset This port remains functional no matter ARMv8-M Security Extensions are included or not When ARMv8-M Security Extensions are not included, all exceptions will use NS vector base address given by this port.
initvtorns	Value	Slave	Reset configuration port - Non-Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset It becomes functional when ARMv8-M Security Extensions are included When ARMv8-M Security Extensions are not included, this port will be ignored.
io_port_in	PVBus	Slave	I/O port pair. See the documentation for the io_port_out port.

Name	Protocol	Type	Description
io_port_out	PVBus	Master	I/O port pair. Used if IOP is true. Transactions from io_port_out which do not "match" should be returned via io_port_in. For performance reasons, the I/O port interface is not modelled directly. Instead, a simple PVBus gasket is inserted at the point in the memory system where the I/O port would be. In hardware, a device would be attached to the port which would tell the CPU whether it would like to intercept each transaction, given its address. This can be modelled in a performant manner by connecting a PVBusMapper-based device to io_port_out which intercepts transactions of interest and passes other transactions back to the CPU via io_port_in. Your I/O port device model is also responsible for aborting transactions which would be aborted on hardware (e.g. exclusives) if necessary.
irq[240]	Signal	Slave	This signal array delivers signals to the NVIC.
lockup	Signal	Master	Asserted when the processor is in lockup state.
niden	Signal	Slave	Non-invasive debug enable, Set=enabled, Clear=disabled
nmi	Signal	Slave	Configure non maskable interrupt.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbus_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
spiden	Signal	Slave	Secure Debug enable , Set=enabled, Clear=disabled
spniden	Signal	Slave	Secure Non-invasive debug enable, Set=enabled, Clear=disabled
stcalib	Value	Slave	This is the calibration value for the Secure (or only, when ARMv8-M Security Extensions are not included) SysTick timer.
stcalibns	Value	Slave	This is the calibration value for the Non-Secure SysTick timer. When ARMv8-M Security Extensions are not included, this port will be ignored.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.
wicenack	Signal	Master	Acknowledge signal for WICENREQ
wicenreq	Signal	Slave	Request for deep sleep to be WIC-based deep sleep.
wicsense[483]	Signal	Master	Indicates which input events can cause the WIC to generate the WAKEUP signal.

Parameters for ARMCortexM23CT

BE

Initialize processor to big endian mode

Type

bool

Default value

0x0

BKPT

Number of breakpoint unit comparators implemented

Type

int

Default value

0x4

CTI

CTI (Cross Trigger Interface) included

Type

bool

Default value

0x0

CTI_irq0_pin

CTI interrupt request 0 pin

Type

int

Default value

0x4

CTI_irq1_pin

CTI interrupt request 1 pin

Type

int

Default value

0x5

DBG

Set whether debug extensions are implemented

Type

bool

Default value

0x1

INITVTOR

Secure vector-table offset at reset

Type

int

Default value

0x0

INITVTORNS

Non-Secure vector-table offset at reset

Type

int

Default value

0x0

IOP

Send all d-side transactions to the port, io_port_out. Transactions which do not match should be returned to the port, io_port_in

Type

bool

Default value

0x0

IRQDIS0

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+0]

Type

int

Default value

0x0

IRQDIS1

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+32]

Type

int

Default value

0x0

IRQDIS2

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+64]

Type

int

Default value

0x0

IRQDIS3

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+96]

Type

int

Default value

0x0

IRQDIS4

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+128]

Type

int

Default value

0x0

IRQDIS5

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+160]

Type

int

Default value

0x0

IRQDIS6

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+192]

Type

int

Default value

0x0

IRQDIS7

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+224]

Type

int

Default value

0x0

MPU_NS

Number of regions in the Non-Secure MPU. If Security Extensions are absent, this is the total number of MPU regions

Type

int

Default value

0x8

MPU_S

Number of regions in the Secure MPU. If Security Extensions are absent, this is ignored

Type

int

Default value

0x8

NUMIRQ

Number of user interrupts

Type

int

Default value

0x10

SAU

Number of SAU regions (0 => no SAU)

Type

int

Default value

0x4

SAU_CTRL.ALLNS

At reset, the SAU treats entire memory space as NS when the SAU is disabled if this is set

Type

bool

Default value

0x0

SAU_CTRL.ENABLE

Enable SAU at reset

Type

bool

Default value

0x0

SAU_REGION0.BADDR

Base address of SAU region0 at reset

Type

int

Default value

0x0

SAU_REGION0.ENABLE

Enable SAU region0 at reset

Type

bool

Default value

0x0

SAU_REGION0.LADDR

Limit address of SAU region0 at reset

Type

int

Default value

0x0

SAU_REGION0.NSC

Set NSC for SAU region0 at reset

Type

bool

Default value

0x0

SAU_REGION1.BADDR

Base address of SAU region1 at reset

Type

int

Default value

0x0

SAU_REGION1.ENABLE

Enable SAU region1 at reset

Type

bool

Default value

0x0

SAU_REGION1.LADDR

Limit address of SAU region1 at reset

Type

int

Default value

0x0

SAU_REGION1.NSC

Set NSC for SAU region1 at reset

Type

bool

Default value

0x0

SAU_REGION2.BADDR

Base address of SAU region2 at reset

Type

int

Default value

0x0

SAU_REGION2.ENABLE

Enable SAU region2 at reset

Type

bool

Default value

0x0

SAU_REGION2.LADDR

Limit address of SAU region2 at reset

Type

int

Default value

0x0

SAU_REGION2.NSC

Set NSC for SAU region2 at reset

Type

bool

Default value

0x0

SAU_REGION3.BADDR

Base address of SAU region3 at reset

Type

int

Default value

0x0

SAU_REGION3.ENABLE

Enable SAU region3 at reset

Type

bool

Default value

0x0

SAU_REGION3.LADDR

Limit address of SAU region3 at reset

Type

int

Default value

0x0

SAU_REGION3.NSC

Set NSC for SAU region3 at reset

Type

bool

Default value

0x0

SAU_REGION4.BADDR

Base address of SAU region4 at reset

Type

int

Default value

0x0

SAU_REGION4.ENABLE

Enable SAU region4 at reset

Type

bool

Default value

0x0

SAU_REGION4.LADDR

Limit address of SAU region4 at reset

Type

int

Default value

0x0

SAU_REGION4.NSC

Set NSC for SAU region4 at reset

Type

bool

Default value

0x0

SAU_REGION5.BADDR

Base address of SAU region5 at reset

Type

int

Default value

0x0

SAU_REGION5.ENABLE

Enable SAU region5 at reset

Type

bool

Default value

0x0

SAU_REGION5.LADDR

Limit address of SAU region5 at reset

Type

int

Default value

0x0

SAU_REGION5.NSC

Set NSC for SAU region5 at reset

Type

bool

Default value

0x0

SAU_REGION6.BADDR

Base address of SAU region6 at reset

Type

int

Default value

0x0

SAU_REGION6.ENABLE

Enable SAU region6 at reset

Type

bool

Default value

0x0

SAU_REGION6.LADDR

Limit address of SAU region6 at reset

Type

int

Default value

0x0

SAU_REGION6.NSC

Set NSC for SAU region6 at reset

Type

bool

Default value

0x0

SAU_REGION7.BADDR

Base address of SAU region7 at reset

Type

int

Default value

0x0

SAU_REGION7.ENABLE

Enable SAU region7 at reset

Type

bool

Default value

0x0

SAU_REGION7.LADDR

Limit address of SAU region7 at reset

Type

int

Default value

0x0

SAU_REGION7.NSC

Set NSC for SAU region7 at reset

Type

bool

Default value

0x0

SECEXT

Whether the ARMv8-M Security Extensions are included

Type

bool

Default value

0x1

SYST

Include SysTick timer functionality (0=Absent, 1=Secure only, 2=Secure and NS)

Type

int

Default value

0x2

VTOR

Include Vector Table Offset Register

Type

bool

Default value

0x1

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

WICLINES

Number of lines supported by the WIC interface

Type

int

Default value

0x12

WPT

Number of watchpoint unit comparators implemented

Type

int

Default value

0x4

cpi_div

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

has_core_dside_bus_gasket

STL gasket enabled

Type

bool

Default value

0x0

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

3.5.56 ARMCortexM33CT

ARMCortexM33CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-410: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexM33CT

This model has the following Iris instances:

Table 3-411: ARMCortexM33CT Iris instances

InstanceName	ComponentName
ARMCortexM33CT	ARM_Cortex-M33
ARMCortexM33CT.acp_mapper	PVBusMapper
ARMCortexM33CT.ext_bus	PVBusLogger
ARMCortexM33CT.ext_bus.mapper	PVBusMapper
ARMCortexM33CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-412: ARMCortexM33CT MTI instances

InstanceName	ComponentName
ARMCortexM33CT	ARM_Cortex-M33
ARMCortexM33CT.acp_mapper	PVBusMapper
ARMCortexM33CT.ext_bus	PVBusLogger
ARMCortexM33CT.ext_bus.mapper	PVBusMapper
ARMCortexM33CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexM33CT contains the following CADI targets:

- ARM_Cortex-M33

About ARMCortexM33CT

The model supports Custom Datapath Extension (CDE). For more information, see [4.7 CDE](#) on page 4148.

Differences between the model and the RTL

The model does not support the following:

- ETM, MTB, CTI, or TPIU. MTB RAM is absent on the model.
- The power control (Q-Channel) interface.

Implementation of ITM in M-class models

This model has a parameter that enables partial support for Instrumentation Trace Macrocell (ITM). In hardware or RTL, trace data from ITM is sent in packets to the trace block serially using a single pin or wire. In the model, if it is enabled, the ITM trace data is output using an MTI trace source

called `ITM`. The `ITM` trace source has an `ITM_PACKET_TYPE` field. The following table shows which packet types the model supports:

Table 3-413: ITM_PACKET_TYPE field values that the model supports

Field value	Description	Supported by model
<code>ITM_SYNC</code>	Synchronization packet	Not supported.
<code>ITM_P_OVERFLOW</code>	Protocol: Overflow packet	Not supported.
<code>ITM_P_LOCAL_TIMESTAMP</code>	Protocol: Local timestamp packets	Not supported.
<code>ITM_P_GLOBAL_TIMESTAMP</code>	Protocol: Global timestamp packets	Not supported.
<code>ITM_P_EXTEN</code>	Protocol: Extension packet	Not supported.
<code>ITM_S_INSTRUMENTATION</code>	Source: Instrumentation packet	Supported.
<code>ITM_S_DWT_EVENT_COUNTER</code>	Hardware source: Event counter wrapping	Not supported.
<code>ITM_S_DWT_EXCEPTION</code>	Hardware source: Exception tracing	Supported.
<code>ITM_S_DWT_PC_SAMPLING</code>	Hardware source: PC sampling	Not supported.
<code>ITM_S_DWT_DATA_PC_TRACE</code>	Hardware source: DWT Data trace PC value	Supported.
<code>ITM_S_DWT_DATA_ADDRESS_TRACE</code>	Hardware source: DWT Data trace address value	Supported.
<code>ITM_S_DWT_DATA_DATA_TRACE</code>	Hardware source: DWT Data trace DATA value	Supported.

Ports for ARMCortexM33CT

Table 3-414: Ports

Name	Protocol	Type	Description
<code>ahbd</code>	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
<code>auxfault</code>	Value	Slave	This is wired to the Auxiliary Fault Status Register.
<code>bigend</code>	Signal	Slave	Configure big endian data format.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions. Referred to in some documents as 'CLKIN'.
<code>coproc_bus</code>	CoprocBusProtocol	Slave	Co-Processor Interface
<code>cpuwait</code>	Signal	Slave	Stall the CPU out of reset
<code>currns</code>	Signal	Master	Current Security state of the processor
<code>currpri</code>	Value	Master	Current execution priority.
<code>dbgen</code>	Signal	Slave	Invasive debug enable
<code>dbgrestart</code>	Signal	Slave	Request for synchronised exit from halt mode
<code>dbgrestarted</code>	Signal	Master	Handshakes with DBGRESTART
<code>edbgrq</code>	Signal	Slave	External request to enter halt mode
<code>event</code>	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
<code>fpxxc</code>	Value	Master	Port that sends the value of the FPxxC exception flags (FPIXC, FPIDC, FPOFC, FPUFC, FPDZC, FPIOC). Each flag is present in the same position as in the FPSCR register (e.g. <code>fpxxc[0]</code> = FPIOC).
<code>halted</code>	Signal	Master	Indicates that the processor is in halt mode
<code>idau</code>	PVBus	Master	The core will generate IDAU requests on this port.

Name	Protocol	Type	Description
idau_invalidate_region	Value_64	Slave	64 bit number to invalid IDAU memory range (start_address<<32 end_address)
initnsvtor	Value	Slave	Reset configuration port - Non-Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset This port remains functional no matter ARMv8-M Security Extensions are included or not When ARMv8-M Security Extensions are not included, all exceptions will use NS vector base address given by this port.
initsvtor	Value	Slave	Reset configuration port - Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset It becomes functional when ARMv8-M Security Extensions are included When ARMv8-M Security Extensions are not included, this port will be ignored.
intnum	Value	Master	Exception number of the current execution context (from IPSR[8:0]) When the processor is in Thread mode, INTNUM is 0 When the processor is in Handler mode, INTNUM is the exception number of the currently-executing exception.
irq[480]	Signal	Slave	This signal array delivers signals to the NVIC.
locknsmpu	Signal	Slave	Disable writes to the Non-Secure MPU_*_NS registers
locknsvtor	Signal	Slave	Cortex-M33-specific LOCKNSVTOR, LOCKSVTAIRCR, LOCKSMPU, LOCKNSMPU, LOCKSAU. Disable writes to VTOR_NS
locksau	Signal	Slave	Disable writes to the SAU_* registers
locksmpu	Signal	Slave	Disable writes to the Secure MPU_* registers
locksvtaircr	Signal	Slave	Disable writes to VTOR_S, AIRCR.PRIS, AIRCR.BFHFNMINS
lockup	Signal	Master	Asserted when the processor is in lockup state.
niden	Signal	Slave	Non-invasive debug enable
nmi	Signal	Slave	Configure non maskable interrupt.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbus_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
spiden	Signal	Slave	Secure invasive debug enable
spniden	Signal	Slave	Secure non-invasive debug enable
stcalib[2]	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.
wicenack	Signal	Master	Acknowledge signal for WICENREQ
wicenreq	Signal	Slave	Request for deep sleep to be WIC-based deep sleep.
wicsense[483]	Signal	Master	Indicates which input events can cause the WIC to generate the WAKEUP signal.

Parameters for ARMCortexM33CT

BIGENDINIT

Initialize processor to big endian mode

Type

bool

Default value

0x0

CDEMAPPEDONCP

Bit N specifies whether the instruction for coprocessor N (CP7:CP0) is redirected to the CDE module

Type

int

Default value

0xff

CDERTLID

Value of ID_AFR0.CDERTLID

Type

int

Default value

0x20

CFGNOCDECP

Bit N means external coprocessor N (CP7:CP0) disable for CDE coprocessor

Type

int

Default value

0x0

CPIF

Specifies whether the external coprocessor interface is included

Type

bool

Default value

0x1

CPNSPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Non-Secure state

Type

int

Default value

0xff

CPSPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Secure state

Type

int

Default value

0xff

CTI

CTI (Cross Trigger Interface) included

Type

bool

Default value

0x0

CTI_irq0_pin

CTI interrupt request 0 pin

Type

int

Default value

0x4

CTI_irq1_pin

CTI interrupt request 1 pin

Type

int

Default value

0x5

DBG_LVL

0: Minimal debug; 1: 2 Watchpoints, 4 Breakpoint comparators; 2: 4 Watchpoints, 8 Breakpoint comparators

Type

int

Default value

0x2

DSP

Set whether the model has the DSP extension

Type

bool

Default value

0x1

FPU

Set whether the model has VFP support

Type

bool

Default value

0x1

INITNSVTOR

Non-Secure vector-table offset at reset

Type

int

Default value

0x0

INITSVTOR

Secure vector-table offset at reset

Type

int

Default value

0x0

IRQDIS0

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+0]

Type

int

Default value

0x0

IRQDIS1

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+32]

Type

int

Default value

0x0

IRQDIS10

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+320]

Type

int

Default value

0x0

IRQDIS11

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+352]

Type

int

Default value

0x0

IRQDIS12

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+384]

Type

int

Default value

0x0

IRQDIS13

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+416]

Type

int

Default value

0x0

IRQDIS14

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+448]

Type

int

Default value

0x0

IRQDIS2

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+64]

Type

int

Default value

0x0

IRQDIS3

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+96]

Type

int

Default value

0x0

IRQDIS4

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+128]

Type

int

Default value

0x0

IRQDIS5

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+160]

Type

int

Default value

0x0

IRQDIS6

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+192]

Type

int

Default value

0x0

IRQDIS7

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+224]

Type

int

Default value

0x0

IRQDIS8

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+256]

Type

int

Default value

0x0

IRQDIS9

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+288]

Type

int

Default value

0x0

IRQLVL

Number of bits of interrupt priority

Type

int

Default value

0x3

ITM

Level of instrumentation trace supported. false : No ITM trace included, true: ITM trace included

Type

bool

Default value

0x1

LOCK_NS_MPU

Lock down of Non-Secure MPU registers write

Type

bool

Default value

0x0

LOCK_SAU

Lock down of SAU registers write

Type

bool

Default value

0x0

LOCK_S_MPU

Lock down of Secure MPU registers write

Type

bool

Default value

0x0

MPU_NS

Number of regions in the Non-Secure MPU. If Security Extensions are absent, this is the total number of MPU regions

Type

int

Default value

0x8

MPU_S

Number of regions in the Secure MPU. If Security Extensions are absent, this is ignored

Type

int

Default value

0x8

NUMIRQ

Number of user interrupts

Type

int

Default value

0x20

SAU

Number of SAU regions (0 => no SAU)

Type

int

Default value

0x4

SAU_CTRL.ALLNS

At reset, the SAU treats entire memory space as NS when the SAU is disabled if this is set

Type

bool

Default value

0x0

SAU_CTRL.ENABLE

Enable SAU at reset

Type

bool

Default value

0x0

SAU_REGION0.BADDR

Base address of SAU region0 at reset

Type

int

Default value

0x0

SAU_REGION0.ENABLE

Enable SAU region0 at reset

Type

bool

Default value

0x0

SAU_REGION0.LADDR

Limit address of SAU region0 at reset

Type

int

Default value

0x0

SAU_REGION0.NSC

Set NSC for SAU region0 at reset

Type

bool

Default value

0x0

SAU_REGION1.BADDR

Base address of SAU region1 at reset

Type

int

Default value

0x0

SAU_REGION1.ENABLE

Enable SAU region1 at reset

Type

bool

Default value

0x0

SAU_REGION1.LADDR

Limit address of SAU region1 at reset

Type

int

Default value

0x0

SAU_REGION1.NSC

Set NSC for SAU region1 at reset

Type

bool

Default value

0x0

SAU_REGION2.BADDR

Base address of SAU region2 at reset

Type

int

Default value

0x0

SAU_REGION2.ENABLE

Enable SAU region2 at reset

Type

bool

Default value

0x0

SAU_REGION2.LADDR

Limit address of SAU region2 at reset

Type

int

Default value

0x0

SAU_REGION2.NSC

Set NSC for SAU region2 at reset

Type

bool

Default value

0x0

SAU_REGION3.BADDR

Base address of SAU region3 at reset

Type

int

Default value

0x0

SAU_REGION3.ENABLE

Enable SAU region3 at reset

Type

bool

Default value

0x0

SAU_REGION3.LADDR

Limit address of SAU region3 at reset

Type

int

Default value

0x0

SAU_REGION3.NSC

Set NSC for SAU region3 at reset

Type

bool

Default value

0x0

SAU_REGION4.BADDR

Base address of SAU region4 at reset

Type

int

Default value

0x0

SAU_REGION4.ENABLE

Enable SAU region4 at reset

Type

bool

Default value

0x0

SAU_REGION4.LADDR

Limit address of SAU region4 at reset

Type

int

Default value

0x0

SAU_REGION4.NSC

Set NSC for SAU region4 at reset

Type

bool

Default value

0x0

SAU_REGION5.BADDR

Base address of SAU region5 at reset

Type

int

Default value

0x0

SAU_REGION5.ENABLE

Enable SAU region5 at reset

Type

bool

Default value

0x0

SAU_REGION5.LADDR

Limit address of SAU region5 at reset

Type

int

Default value

0x0

SAU_REGION5.NSC

Set NSC for SAU region5 at reset

Type

bool

Default value

0x0

SAU_REGION6.BADDR

Base address of SAU region6 at reset

Type

int

Default value

0x0

SAU_REGION6.ENABLE

Enable SAU region6 at reset

Type

bool

Default value

0x0

SAU_REGION6.LADDR

Limit address of SAU region6 at reset

Type

int

Default value

0x0

SAU_REGION6.NSC

Set NSC for SAU region6 at reset

Type

bool

Default value

0x0

SAU_REGION7.BADDR

Base address of SAU region7 at reset

Type

int

Default value

0x0

SAU_REGION7.ENABLE

Enable SAU region7 at reset

Type

bool

Default value

0x0

SAU_REGION7.LADDR

Limit address of SAU region7 at reset

Type

int

Default value

0x0

SAU_REGION7.NSC

Set NSC for SAU region7 at reset

Type

bool

Default value

0x0

SECEXT

Whether the ARMv8-M Security Extensions are included

Type

bool

Default value

0x1

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

WICLINES

Number of lines supported by the WIC interface

Type

int

Default value

0x23

cde_impl_name

Name of the CDE implementation for this core (implementation contributed by MTI plugin).

Type

string

Default value**cpi_div**

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

has_cde

Enables Custom Datapath Extensions

Type

bool

Default value

0x0

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CPUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CPUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

3.5.57 ARMCortexM35PCT

ARMCortexM35PCT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-415: IP revisions support

Revision	Quality level
r1p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexM35PCT

This model has the following Iris instances:

Table 3-416: ARMCortexM35PCT Iris instances

InstanceName	ComponentName
ARMCortexM35PCT	ARM_Cortex-M35P
ARMCortexM35PCT.acp_mapper	PVBusMapper
ARMCortexM35PCT.ext_bus	PVBusLogger
ARMCortexM35PCT.ext_bus.mapper	PVBusMapper
ARMCortexM35PCT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-417: ARMCortexM35PCT MTI instances

InstanceName	ComponentName
ARMCortexM35PCT	ARM_Cortex-M35P
ARMCortexM35PCT.acp_mapper	PVBusMapper
ARMCortexM35PCT.ext_bus	PVBusLogger
ARMCortexM35PCT.ext_bus.mapper	PVBusMapper
ARMCortexM35PCT.l2_flusher	AsyncCacheFlushUnit

ARMCortexM35PCT contains the following CADI targets:

- ARM_Cortex-M35P

Differences between the model and the RTL

- The model does not support the following:
 - ETM, MTB, CTI, or TPIU. MTB RAM is absent on the model.
 - Caches.
 - The co-processor interface.
 - The power control (Q-Channel) interface.

- The model does not implement any physical security features.
- Bits[3:0] of the Anti-tampering Features Control Register are supported for read/write. No functionality is implemented.
- Read/write access to the Anti-tampering Features Control Register is supported using SECKEY. No functionality is implemented.

Implementation of ITM in M-class models

This model has a parameter that enables partial support for Instrumentation Trace Macrocell (ITM). In hardware or RTL, trace data from ITM is sent in packets to the trace block serially using a single pin or wire. In the model, if it is enabled, the ITM trace data is output using an MTI trace source called `ITM`. The `ITM` trace source has an `ITM_PACKET_TYPE` field. The following table shows which packet types the model supports:

Table 3-418: ITM_PACKET_TYPE field values that the model supports

Field value	Description	Supported by model
<code>ITM_SYNC</code>	Synchronization packet	Not supported.
<code>ITM_P_OVERFLOW</code>	Protocol: Overflow packet	Not supported.
<code>ITM_P_LOCAL_TIMESTAMP</code>	Protocol: Local timestamp packets	Not supported.
<code>ITM_P_GLOBAL_TIMESTAMP</code>	Protocol: Global timestamp packets	Not supported.
<code>ITM_P_EXTEN</code>	Protocol: Extension packet	Not supported.
<code>ITM_S_INSTRUMENTATION</code>	Source: Instrumentation packet	Supported.
<code>ITM_S_DWT_EVENT_COUNTER</code>	Hardware source: Event counter wrapping	Not supported.
<code>ITM_S_DWT_EXCEPTION</code>	Hardware source: Exception tracing	Supported.
<code>ITM_S_DWT_PC_SAMPLING</code>	Hardware source: PC sampling	Not supported.
<code>ITM_S_DWT_DATA_PC_TRACE</code>	Hardware source: DWT Data trace PC value	Supported.
<code>ITM_S_DWT_DATA_ADDRESS_TRACE</code>	Hardware source: DWT Data trace address value	Supported.
<code>ITM_S_DWT_DATA_DATA_TRACE</code>	Hardware source: DWT Data trace DATA value	Supported.

Ports for ARMCortexM35PCT

Table 3-419: Ports

Name	Protocol	Type	Description
<code>auxfault</code>	Value	Slave	This is wired to the Auxiliary Fault Status Register.
<code>bigend</code>	Signal	Slave	Configure big endian data format.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions. Referred to in some documents as 'CLKIN'.
<code>coproc_bus</code>	CoprocBusProtocol	Slave	Co-Processor Interface
<code>cpuwait</code>	Signal	Slave	Stall the CPU out of reset
<code>currns</code>	Signal	Master	Current Security state of the processor
<code>currpri</code>	Value	Master	Current execution priority.
<code>dbgen</code>	Signal	Slave	Invasive debug enable
<code>dbgrestart</code>	Signal	Slave	Request for synchronised exit from halt mode

Name	Protocol	Type	Description
dbgrestarted	Signal	Master	Handshakes with DBGRESTART
edbgrq	Signal	Slave	External request to enter halt mode
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
fpxxc	Value	Master	Port that sends the value of the FPxxC exception flags (FPIXC, FPIDC, FPOFC, FPUFC, FPDZC, FPIOC). Each flag is present in the same position as in the FPSCR register (e.g. fpxxc[0] = FPIOC).
halted	Signal	Master	Indicates that the processor is in halt mode
idau	PVBus	Master	The core will generate IDAU requests on this port.
idau_invalidate_region	Value_64	Slave	64 bit number to invalid IDAU memory range (start_address<<32 end_address)
initnsvtor	Value	Slave	Reset configuration port - Non-Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset This port remains functional no matter ARMv8-M Security Extensions are included or not When ARMv8-M Security Extensions are not included, all exceptions will use NS vector base address given by this port.
initsvtor	Value	Slave	Reset configuration port - Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset It becomes functional when ARMv8-M Security Extensions are included When ARMv8-M Security Extensions are not included, this port will be ignored.
intnum	Value	Master	Exception number of the current execution context (from IPSR[8:0]) When the processor is in Thread mode, INTNUM is 0 When the processor is in Handler mode, INTNUM is the exception number of the currently-executing exception.
irq[480]	Signal	Slave	This signal array delivers signals to the NVIC.
LOCKATFCR	Signal	Slave	Port Lock ATFCR register
locknsmpu	Signal	Slave	Disable writes to the Non-Secure MPU_*_NS registers
locknsvtor	Signal	Slave	Disable writes to VTOR_NS
locksau	Signal	Slave	Disable writes to the SAU_* registers
locksmpu	Signal	Slave	Disable writes to the Secure MPU_* registers
locksvtaircr	Signal	Slave	Disable writes to VTOR_S, AIRCR.PRIS, AIRCR.BFHFNMINS
lockup	Signal	Master	Asserted when the processor is in lockup state.
niden	Signal	Slave	Non-invasive debug enable
nmi	Signal	Slave	Configure non maskable interrupt.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbus_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
spiden	Signal	Slave	Secure invasive debug enable
spniden	Signal	Slave	Secure non-invasive debug enable
stcalib[2]	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.

Name	Protocol	Type	Description
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.
wicenack	Signal	Master	Acknowledge signal for WICENREQ
wicenreq	Signal	Slave	Request for deep sleep to be WIC-based deep sleep.
wicsense[483]	Signal	Master	Indicates which input events can cause the WIC to generate the WAKEUP signal.

Parameters for ARMCortexM35PCT

ATFINITEN

ATFCR is enabled when the core goes out of reset

Type

bool

Default value

0x0

BIGENDINIT

Initialize processor to big endian mode

Type

bool

Default value

0x0

CPIF

Specifies whether the external coprocessor interface is included

Type

bool

Default value

0x1

CPNSPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Non-Secure state

Type

int

Default value

0xff

CPSPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Secure state

Type

int

Default value

0xff

CTI

CTI (Cross Trigger Interface) included

Type

bool

Default value

0x0

CTI_irq0_pin

CTI interrupt request 0 pin

Type

int

Default value

0x4

CTI_irq1_pin

CTI interrupt request 1 pin

Type

int

Default value

0x5

DBG_LVL

0: Minimal debug; 1: 2 Watchpoints, 4 Breakpoint comparators; 2: 4 Watchpoints, 8 Breakpoint comparators

Type

int

Default value

0x2

DSP

Set whether the model has the DSP extension

Type

bool

Default value

0x1

FPU

Set whether the model has VFP support

Type

bool

Default value

0x1

INITNSVTOR

Non-Secure vector-table offset at reset

Type

int

Default value

0x0

INITSVTOR

Secure vector-table offset at reset

Type

int

Default value

0x0

IRQDIS0

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+0]

Type

int

Default value

0x0

IRQDIS1

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+32]

Type

int

Default value

0x0

IRQDIS10

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+320]

Type

int

Default value

0x0

IRQDIS11

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+352]

Type

int

Default value

0x0

IRQDIS12

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+384]

Type

int

Default value

0x0

IRQDIS13

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+416]

Type

int

Default value

0x0

IRQDIS14

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+448]

Type

int

Default value

0x0

IRQDIS2

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+64]

Type

int

Default value

0x0

IRQDIS3

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+96]

Type

int

Default value

0x0

IRQDIS4

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+128]

Type

int

Default value

0x0

IRQDIS5

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+160]

Type

int

Default value

0x0

IRQDIS6

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+192]

Type

int

Default value

0x0

IRQDIS7

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+224]

Type

int

Default value

0x0

IRQDIS8

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+256]

Type

int

Default value

0x0

IRQDIS9

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+288]

Type

int

Default value

0x0

IRQVLV

Number of bits of interrupt priority

Type

int

Default value

0x3

ITM

Level of instrumentation trace supported. false : No ITM trace included, true: ITM trace included

Type

bool

Default value

0x1

LOCK_NS_MPU

Lock down of Non-Secure MPU registers write

Type

bool

Default value

0x0

LOCK_SAU

Lock down of SAU registers write

Type

bool

Default value

0x0

LOCK_S_MPU

Lock down of Secure MPU registers write

Type

bool

Default value

0x0

MPU_NS

Number of regions in the Non-Secure MPU. If Security Extensions are absent, this is the total number of MPU regions

Type

int

Default value

0x8

MPU_S

Number of regions in the Secure MPU. If Security Extensions are absent, this is ignored

Type

int

Default value

0x8

NUMIRQ

Number of user interrupts

Type

int

Default value

0x20

SAU

Number of SAU regions (0 => no SAU)

Type

int

Default value

0x4

SAU_CTRL.ALLNS

At reset, the SAU treats entire memory space as NS when the SAU is disabled if this is set

Type

bool

Default value

0x0

SAU_CTRL.ENABLE

Enable SAU at reset

Type

bool

Default value

0x0

SAU_REGION0.BADDR

Base address of SAU region0 at reset

Type

int

Default value

0x0

SAU_REGION0.ENABLE

Enable SAU region0 at reset

Type

bool

Default value

0x0

SAU_REGION0.LADDR

Limit address of SAU region0 at reset

Type

int

Default value

0x0

SAU_REGION0.NSC

Set NSC for SAU region0 at reset

Type

bool

Default value

0x0

SAU_REGION1.BADDR

Base address of SAU region1 at reset

Type

int

Default value

0x0

SAU_REGION1.ENABLE

Enable SAU region1 at reset

Type

bool

Default value

0x0

SAU_REGION1.LADDR

Limit address of SAU region1 at reset

Type

int

Default value

0x0

SAU_REGION1.NSC

Set NSC for SAU region1 at reset

Type

bool

Default value

0x0

SAU_REGION2.BADDR

Base address of SAU region2 at reset

Type

int

Default value

0x0

SAU_REGION2.ENABLE

Enable SAU region2 at reset

Type

bool

Default value

0x0

SAU_REGION2.LADDR

Limit address of SAU region2 at reset

Type

int

Default value

0x0

SAU_REGION2.NSC

Set NSC for SAU region2 at reset

Type

bool

Default value

0x0

SAU_REGION3.BADDR

Base address of SAU region3 at reset

Type

int

Default value

0x0

SAU_REGION3.ENABLE

Enable SAU region3 at reset

Type

bool

Default value

0x0

SAU_REGION3.LADDR

Limit address of SAU region3 at reset

Type

int

Default value

0x0

SAU_REGION3.NSC

Set NSC for SAU region3 at reset

Type

bool

Default value

0x0

SAU_REGION4.BADDR

Base address of SAU region4 at reset

Type

int

Default value

0x0

SAU_REGION4.ENABLE

Enable SAU region4 at reset

Type

bool

Default value

0x0

SAU_REGION4.LADDR

Limit address of SAU region4 at reset

Type

int

Default value

0x0

SAU_REGION4.NSC

Set NSC for SAU region4 at reset

Type

bool

Default value

0x0

SAU_REGION5.BADDR

Base address of SAU region5 at reset

Type

int

Default value

0x0

SAU_REGION5.ENABLE

Enable SAU region5 at reset

Type

bool

Default value

0x0

SAU_REGION5.LADDR

Limit address of SAU region5 at reset

Type

int

Default value

0x0

SAU_REGION5.NSC

Set NSC for SAU region5 at reset

Type

bool

Default value

0x0

SAU_REGION6.BADDR

Base address of SAU region6 at reset

Type

int

Default value

0x0

SAU_REGION6.ENABLE

Enable SAU region6 at reset

Type

bool

Default value

0x0

SAU_REGION6.LADDR

Limit address of SAU region6 at reset

Type

int

Default value

0x0

SAU_REGION6.NSC

Set NSC for SAU region6 at reset

Type

bool

Default value

0x0

SAU_REGION7.BADDR

Base address of SAU region7 at reset

Type

int

Default value

0x0

SAU_REGION7.ENABLE

Enable SAU region7 at reset

Type

bool

Default value

0x0

SAU_REGION7.LADDR

Limit address of SAU region7 at reset

Type

int

Default value

0x0

SAU_REGION7.NSC

Set NSC for SAU region7 at reset

Type

bool

Default value

0x0

SECEXT

Whether the ARMv8-M Security Extensions are included

Type

bool

Default value

0x1

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

WICLINES

Number of lines supported by the WIC interface

Type

int

Default value

0x23

cpi_div

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

3.5.58 ARMCortexM52CT

CortexM52CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-420: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters removed:

- BEATS_PER_TICK
- BF_is_nop
- ID_ISAR0.CmpBranch
- aircr_iesb_is_writable
- aircr_iesb_reset
- has_unpriviledged_debug

- num_pmu_counters
- ras_ERRFR0
- ras_cei_pin
- ras_cei_support
- ras_eri_pin
- ras_eri_support
- ras_error_record
- ras_fhi_pin
- ras_fhi_support

Iris and MTI instances for ARM CortexM52CT

This model has the following Iris instances:

Table 3-421: ARM CortexM52CT Iris instances

InstanceName	ComponentName
ARM CortexM52CT	ARM_Cortex-M52
ARM CortexM52CT.acp_mapper	PVBusMapper
ARM CortexM52CT.ext_bus	PVBusLogger
ARM CortexM52CT.ext_bus.mapper	PVBusMapper
ARM CortexM52CT.l1_incoherent_interconnect	PVCache
ARM CortexM52CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARM CortexM52CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARM CortexM52CT.l1dcache	PVCache

InstanceName	ComponentName
ARMCortexM52CT.11dcache.upstream[0]	PVBusSlave
ARMCortexM52CT.11icache	PVCache
ARMCortexM52CT.11icache.upstream[0]	PVBusSlave
ARMCortexM52CT.12_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-422: ARMCortexM52CT MTI instances

InstanceName	ComponentName
ARMCortexM52CT	ARM_Cortex-M52
ARMCortexM52CT.acp_mapper	PVBusMapper
ARMCortexM52CT.ext_bus	PVBusLogger
ARMCortexM52CT.ext_bus.mapper	PVBusMapper
ARMCortexM52CT.11_incoherent_interconnect	PVCache
ARMCortexM52CT.11_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexM52CT.11_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexM52CT.11dcache	PVCache
ARMCortexM52CT.11dcache.upstream[0]	PVBusSlave
ARMCortexM52CT.11icache	PVCache
ARMCortexM52CT.11icache.upstream[0]	PVBusSlave
ARMCortexM52CT.12_flusher	AsyncCacheFlushUnit

ARMCortexM52CT contains the following CADI targets:

- ARM_Cortex-M52

About ARMCortexM52CT

This model supports the M-Profile Vector Extension (MVE) and the Custom Datapath Extension (CDE). For more information, see [4.7 CDE](#) on page 4148.

Ports for ARMCortexM52CT

Table 3-423: Ports

Name	Protocol	Type	Description
ahbd	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
ahbp_m	PVBus	Master	The core will generate Vendor System data accesses on this port.
ahbs	PVBus	Slave	Slave AHB - External master (e.g. DMA) can write to TCMs (whether or not enabled in xTCMCR)
auxfault	Value	Slave	This is wired to the Auxiliary Fault Status Register.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions. Referred to in some documents as 'CLKIN'.
coproc_bus	CoprocBusProtocol	Slave	Co-Processor Interface
core_dside_bus_gasket_in	PVBus	Slave	-
core_dside_bus_gasket_out	PVBus	Master	-
cpuwait	Signal	Slave	Stall the CPU out of reset
currns	Signal	Master	Current Security state of the processor
currpri	Value	Master	Current execution priority.
dbgen	Signal	Slave	Invasive debug enable
dbgrestart	Signal	Slave	Request for synchronised exit from halt mode
dbgrestarted	Signal	Master	Handshakes with DBGRESTART
edbgrq	Signal	Slave	External request to enter halt mode
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
fpxxc	Value	Master	Port that sends the value of the FPxxC exception flags (FPIXC, FPIDC, FPOFC, FPUFC, FPDZC, FPIOC). Each flag is present in the same position as in the FPSCR register (e.g. fpxxc[0] = FPIOC).
halted	Signal	Master	Indicates that the processor is in halt mode
idau	PVBus	Master	The core will generate IDAU requests on this port.
idau_invalidate_region	Value_64	Slave	64 bit number to invalid IDAU memory ranage (start_address<<32 end_address)
initnsvtor	Value	Slave	Reset configuration port - Non-Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset This port remains functional no matter ARMv8-M Security Extensions are included or not When ARMv8-M Security Extensions are not included, all exceptions will use NS vector base address given by this port.
initpahben	Signal	Slave	Enable P-AHB on the next reset

Name	Protocol	Type	Description
initsvtor	Value	Slave	Reset configuration port - Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset It becomes functional when ARMv8-M Security Extensions are included When ARMv8-M Security Extensions are not included, this port will be ignored.
inittcmem[2]	Signal	Slave	Reset configuration port - TCM enable initialisation out of reset Bit[0] HIGH = ITCM is enabled Bit[1] HIGH = DTCM is enabled
intnum	Value	Master	Exception number of the current execution context (from IPSR[8:0]) When the processor is in Thread mode, INTNUM is 0 When the processor is in Handler mode, INTNUM is the exception number of the currently-executing exception.
irq[480]	Signal	Slave	This signal array delivers signals to the NVIC.
lockdcaic	Signal	Slave	Disable access to instruction cache direct cache access registers
lockdtgu	Signal	Slave	Disable writes to registers associated with the DTGU
lockitgu	Signal	Slave	Disable writes to registers associated with the ITGU
locknsmpu	Signal	Slave	Disable writes to the Non-Secure MPU_*_NS registers
locknsvtor	Signal	Slave	Cortex-M52-specific LOCKNSVTOR, LOCKSVTAIRCR, LOCKSMPU, LOCKNSMPU, LOCKSAU. Disable writes to VTOR_NS
lockpahb	Signal	Slave	P-AHB related ports Disable writes to PAHBCR
locksau	Signal	Slave	Disable writes to the SAU_* registers
locksmpu	Signal	Slave	Disable writes to the Secure MPU_* registers
locksvtaircr	Signal	Slave	Disable writes to VTOR_S, AIRCR.PRIS, AIRCR.BFHFNMINS
locktcm	Signal	Slave	Disable writes to registers associated with the TCM
lockup	Signal	Master	Asserted when the processor is in lockup state.
niden	Signal	Slave	Non-invasive debug enable
nmi	Signal	Slave	Configure non maskable interrupt.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
prequest	PChannel	Slave	Low Power Interface
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pdbus_m	PVBus	Master	The core will generate bus requests on this port.
qrequest	PChannel	Slave	-
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
spiden	Signal	Slave	Secure invasive debug enable
spniden	Signal	Slave	Secure non-invasive debug enable
stcalib[2]	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.
wicenack	Signal	Master	Acknowledge signal for WICENREQ

Name	Protocol	Type	Description
wicenreq	Signal	Slave	Request for deep sleep to be WIC-based deep sleep.
wicsense[483]	Signal	Master	Indicates which input events can cause the WIC to generate the WAKEUP signal.

Parameters for ARM CortexM52CT

CDEMAPPEDONCP

Bit N specifies whether the instruction for coprocessor N (CP7:CP0) is redirected to the CDE module

Type

int

Default value

0xff

CDERTLID

Value of ID_AFR0.CDERTLID

Type

int

Default value

0x20

CFGBIGEND

Initialize processor to big endian mode

Type

bool

Default value

0x0

CFGCPUINST

CPU instance number. This is part of the TCM base address, in bits 25:24.

Type

int

Default value

0x0

CFGDTCMSZ

Size of the data TCM. 0=No DTCM implemented. Otherwise=Size of DTCM=pow(2, CFGDTCMSZ - 1) KB. Minimum size is 4KB

Type

int

Default value

0x9

CFGITCMSZ

Size of the instruction TCM. 0=No ITCM implemented. Otherwise=Size of ITCM=pow(2, CFGITCMSZ - 1) KB. Minimum size is 4KB

Type

int

Default value

0x9

CFGMEMALIAS

Memory address alias bit for the ITCM, DTCM and P-AHB regions. 0=No alias, 1=Alias bit 26, 2=Alias bit 27, 4=Alias bit 28

Type

int

Default value

0x0

CFGNOCDECP

Bit N means external coprocessor N (CP7:CP0) disable for CDE coprocessor

Type

int

Default value

0x0

CFGPACBTI

Enables support for the Pointer Authentication and Branch Target Identification (PACBTI) Extension. FALSE: Disabled, TRUE:PAC implemented using the QARMA3 algorithm with BTI

Type

bool

Default value

0x0

CFGPAHBSZ

Size of the P-AHB peripheral port memory region. 0=P-AHB disabled, 1=64MB, 2=128MB, 3=256MB, 4=512MB

Type

int

Default value

0x0

CPIF

Specifies whether the external coprocessor interface is included

Type

bool

Default value

0x1

CPNSPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Non-Secure state

Type

int

Default value

0xff

CPSPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Secure state

Type

int

Default value

0xff

CTI

CTI (Cross Trigger Interface) included

Type

bool

Default value

0x0

CTI_irq0_pin

CTI interrupt request 0 pin

Type

int

Default value

0x4

CTI_irq1_pin

CTI interrupt request 1 pin

Type

int

Default value

0x5

DBGVLVL

0: Minimal debug; 1: 2 Watchpoints, 4 Breakpoint comparators; 2: 4 Watchpoints, 8 Breakpoint comparators; 3: 8 Watchpoints, 8 Breakpoint comparators

Type

int

Default value

0x2

DCACHESZ

Whether the D-cache is included and, if included, the size of it. Bit 0: 0=No D-cache included, 1=D-cache included. Bits [4:1]: 0x0=4KB D-cache, 0x1=8KB D-cache, 0x3=16KB D-cache, 0x7=32KB D-cache, 0xF=64KB D-cache

Type

int

Default value

0xf

DTGU

DTCM Security Gate Unit included

Type

bool

Default value

0x0

DTGUBLKSZ

DTCM gate unit block size. Size= $\text{pow}(2, \text{DTGUBLKSZ} + 5)$ bytes

Type

int

Default value

0x3

DTGUMAXBLKS

Maximum number of DTCM gate unit blocks. Number of blocks= $\text{pow}(2, \text{DTGUMAXBLKS})$

Type

int

Default value

0x0

ECOREVNUM

ECO Revision number

Type

int

Default value

0x0

ERRDEVID.NUM

RAS: Number of implemented error record indexes, 0 to 1.

Type

int

Default value

0x1

ETM

Support for ETM trace. false : No ETM trace included, true: ETM trace included

Type

bool

Default value

0x1

FPMVE

Set whether the model has FP and / or MVE support. 0: No FP and MVE support. 1: FP half and single precision. 2: FP half, single and double precision. 3: MVE integer. 4: FP half and single precision and MVE integer. 5: FP half, single and double precision and MVE floating point.

Type

int

Default value

0x5

ICACHESZ

Whether the I-cache is included and, if included, the size of it. Bit 0: 0=No I-cache included, 1=I-cache included. Bits [6:1]: 0x0=1KB I-cache (only with unified cache), 0x1=2KB I-cache (only with unified cache), 0x3=4KB I-cache, 0x7=8KB I-cache, 0xF=16KB I-cache, 0x1F=32KB I-cache, 0x3F=64KB I-cache

Type

int

Default value

0x3f

INITNSVTOR

Non-Secure vector-table offset at reset

Type

int

Default value

0x0

INITPAHBEN

The P-AHB enable state at reset

Type

bool

Default value

0x0

INITSVTOR

Secure vector-table offset at reset

Type

int

Default value

0x0

INITTCMEN

The TCM enable state at reset. Bit 0 corresponds to ITCM enable state, bit 1 corresponds to DTCM enable state

Type

int

Default value

0x3

IRQDIS0

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+0]

Type

int

Default value

0x0

IRQDIS1

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+32]

Type

int

Default value

0x0

IRQDIS10

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+320]

Type

int

Default value

0x0

IRQDIS11

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+352]

Type

int

Default value

0x0

IRQDIS12

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+384]

Type

int

Default value

0x0

IRQDIS13

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+416]

Type

int

Default value

0x0

IRQDIS14

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+448]

Type

int

Default value

0x0

IRQDIS2

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+64]

Type

int

Default value

0x0

IRQDIS3

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+96]

Type

int

Default value

0x0

IRQDIS4

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+128]

Type

int

Default value

0x0

IRQDIS5

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+160]

Type

int

Default value

0x0

IRQDIS6

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+192]

Type

int

Default value

0x0

IRQDIS7

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+224]

Type

int

Default value

0x0

IRQDIS8

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+256]

Type

int

Default value

0x0

IRQDIS9

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+288]

Type

int

Default value

0x0

IRQLVL

Number of bits of interrupt priority

Type

int

Default value

0x3

ITGU

ITCM Security Gate Unit included

Type

bool

Default value

0x0

ITGUBLKSZ

ITCM gate unit block size. Size=pow(2, ITGUBLKSZ + 5) bytes

Type

int

Default value

0x3

ITGUMAXBLKS

Maximum number of ITCM gate unit blocks. Number of blocks=pow(2, ITGUMAXBLKS)

Type

int

Default value

0x0

ITM

Level of instrumentation trace supported. false : No ITM trace included, true: ITM trace included

Type

bool

Default value

0x1

IWIC

Include support for Internal Wake-up Interrupt Controller

Type

bool

Default value

0x1

LOCKDTGU

Lock down of Data TGU registers write

Type

bool

Default value

0x0

LOCKITGU

Lock down of Instruction TGU registers write

Type

bool

Default value

0x0

LOCKTCM

Lock down of TCM registers write

Type

bool

Default value

0x0

LOCK_NS_MPU

Lock down of Non-Secure MPU registers write

Type

bool

Default value

0x0

LOCK_SAU

Lock down of SAU registers write

Type

bool

Default value

0x0

LOCK_S_MPU

Lock down of Secure MPU registers write

Type

bool

Default value

0x0

MPU_NS

Number of regions in the Non-Secure MPU. If Security Extensions are absent, this is the total number of MPU regions

Type

int

Default value

0x8

MPU_S

Number of regions in the Secure MPU. If Security Extensions are absent, this is ignored

Type

int

Default value

0x8

NUMIRQ

Number of user interrupts

Type

int

Default value

0x20

SAU

Number of SAU regions (0 => no SAU)

Type

int

Default value

0x4

SECEXT

Whether the ARMv8-M Security Extensions are included

Type

bool

Default value

0x1

UCACHE

Whether the I-cache acts as a unified cache (ICACHESZ is used for the size)

Type

bool

Default value

0x0

WICLINES

Number of lines supported by the WIC interface

Type

int

Default value

0x23

cde_impl_name

Name of the CDE implementation for this core (implementation contributed by MTI plugin).

Type

string

Default value**cpi_div**

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

delay_faultmask_update

Delay FAULTMASK update to context sync

Type

bool

Default value

0x0

delay_sysreg_update

Delay some system register updates (e.g. SHCSR) to context sync

Type

bool

Default value

0x0

ecc_on

Enable Error Correcting Code

Type

bool

Default value

0x0

has_cde

Enables Custom Datapath Extensions

Type

bool

Default value

0x0

has_core_dside_bus_gasket

STL gasket enabled

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

trace_style

MVE instruction trace style: Add 16 for [**-] beat trace. Add 32 for tracing IMPLIED LOB instructions. Add 64 to change opcode of implied BF to 0xBF00

Type

int

Default value

0x2

vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

3.5.59 ARMCortexM55CT

CortexM55CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-424: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters removed:

- BEATS_PER_TICK
- BF_is_nop
- ID_ISAR0.CmpBranch
- aircr_iesb_is_writable
- aircr_iesb_reset
- has_unpriviledged_debug
- num_pmu_counters
- ras_ERRFR0
- ras_cei_pin
- ras_cei_support
- ras_eri_pin
- ras_eri_support
- ras_error_record
- ras_fhi_pin
- ras_fhi_support

Iris and MTI instances for ARM CortexM55CT

This model has the following Iris instances:

Table 3-425: ARM CortexM55CT Iris instances

InstanceName	ComponentName
ARM CortexM55CT	ARM_Cortex-M55
ARM CortexM55CT.acp_mapper	PVBusMapper
ARM CortexM55CT.ext_bus	PVBusLogger
ARM CortexM55CT.ext_bus.mapper	PVBusMapper
ARM CortexM55CT.l1_incoherent_interconnect	PVCache
ARM CortexM55CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARM CortexM55CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave

InstanceName	ComponentName
ARMCortexM55CT.11_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexM55CT.11dcache	PVCache
ARMCortexM55CT.11dcache.upstream[0]	PVBusSlave
ARMCortexM55CT.11icache	PVCache
ARMCortexM55CT.11icache.upstream[0]	PVBusSlave
ARMCortexM55CT.12_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-426: ARMCortexM55CT MTI instances

InstanceName	ComponentName
ARMCortexM55CT	ARM_Cortex-M55
ARMCortexM55CT.acp_mapper	PVBusMapper
ARMCortexM55CT.ext_bus	PVBusLogger
ARMCortexM55CT.ext_bus.mapper	PVBusMapper
ARMCortexM55CT.11_incoherent_interconnect	PVCache
ARMCortexM55CT.11_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexM55CT.11_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexM55CT.11dcache	PVCache

InstanceName	ComponentName
ARMCortexM55CT.l1dcache.upstream[0]	PVBusSlave
ARMCortexM55CT.l1icache	PVCache
ARMCortexM55CT.l1icache.upstream[0]	PVBusSlave
ARMCortexM55CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexM55CT contains the following CADI targets:

- ARM_Cortex-M55

About ARMCortexM55CT

The model supports Custom Datapath Extension (CDE). For more information, see [4.7 CDE](#) on page 4148.

The model does not support the following functionality:

- Cross Trigger Interface (CTI).
- Programmable MBIST controller (PMC-100).
- Error Correcting Code (ECC).
- Q-Channel.

The following interfaces and registers are not modeled:

- ITM and ETM trace and trace synchronization and trigger interface signals.
- Dual-core lock-step operation.
- Interrupt latencies.
- Memory System Control Register (MSCR).
- Prefetcher Control Register (PFCR).
- Direct cache access registers.

Differences between the model and the RTL

In hardware, `PMU_CCNT` is an alias of the `DWT_CYCCNT` register, so they contain the same values. In the model, `PMU_CCNT` is implemented differently to `DWT_CYCCNT`, so they contain different values. The value held in `DWT_CYCCNT` is not representative of hardware.

Implementation of ITM in M-class models

This model has a parameter that enables partial support for Instrumentation Trace Macrocell (ITM). In hardware or RTL, trace data from ITM is sent in packets to the trace block serially using a single pin or wire. In the model, if it is enabled, the ITM trace data is output using an MTI trace source called `ITM`. The `ITM` trace source has an `ITM_PACKET_TYPE` field. The following table shows which packet types the model supports:

Table 3-427: ITM_PACKET_TYPE field values that the model supports

Field value	Description	Supported by model
ITM_SYNC	Synchronization packet	Not supported.
ITM_P_OVERFLOW	Protocol: Overflow packet	Not supported.
ITM_P_LOCAL_TIMESTAMP	Protocol: Local timestamp packets	Not supported.
ITM_P_GLOBAL_TIMESTAMP	Protocol: Global timestamp packets	Not supported.
ITM_P_EXTEN	Protocol: Extension packet	Not supported.
ITM_S_INSTRUMENTATION	Source: Instrumentation packet	Supported.
ITM_S_DWT_EVENT_COUNTER	Hardware source: Event counter wrapping	Not supported.
ITM_S_DWT_EXCEPTION	Hardware source: Exception tracing	Supported.
ITM_S_DWT_PC_SAMPLING	Hardware source: PC sampling	Not supported.
ITM_S_DWT_DATA_PC_TRACE	Hardware source: DWT Data trace PC value	Supported.
ITM_S_DWT_DATA_ADDRESS_TRACE	Hardware source: DWT Data trace address value	Supported.
ITM_S_DWT_DATA_DATA_TRACE	Hardware source: DWT Data trace DATA value	Supported.

Ports for ARMCortexM55CT

Table 3-428: Ports

Name	Protocol	Type	Description
ahbd	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
ahbp_m	PVBus	Master	The core will generate Vendor System data accesses on this port.
ahbs	PVBus	Slave	Slave AHB - External master (e.g. DMA) can write to TCMs (whether or not enabled in xTCMCR)
auxfault	Value	Slave	This is wired to the Auxiliary Fault Status Register.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions. Referred to in some documents as 'CLKIN'.
coproc_bus	CoprocBusProtocol	Slave	Co-Processor Interface
core_dside_bus_gasket_in	PVBus	Slave	-
core_dside_bus_gasket_out	PVBus	Master	-
cpuwait	Signal	Slave	Stall the CPU out of reset
currns	Signal	Master	Current Security state of the processor
currpri	Value	Master	Current execution priority.
dbgen	Signal	Slave	Invasive debug enable
dbgrestart	Signal	Slave	Request for synchronised exit from halt mode
dbgrestarted	Signal	Master	Handshakes with DBGRESTART
edbgrq	Signal	Slave	External request to enter halt mode
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.

Name	Protocol	Type	Description
fpxxc	Value	Master	Port that sends the value of the FPxxC exception flags (FPIX, FPIDC, FPOFC, FPUFC, FPDZC, FPIOC). Each flag is present in the same position as in the FPSCR register (e.g. fpxxc[0] = FPIOC).
halted	Signal	Master	Indicates that the processor is in halt mode
idau	PVBus	Master	The core will generate IDAU requests on this port.
idau_invalidate_region	Value_64	Slave	64 bit number to invalid IDAU memory range (start_address<<32 end_address)
initnsvtor	Value	Slave	Reset configuration port - Non-Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset This port remains functional no matter ARMv8-M Security Extensions are included or not When ARMv8-M Security Extensions are not included, all exceptions will use NS vector base address given by this port.
initpahben	Signal	Slave	Enable P-AHB on the next reset
initsvtor	Value	Slave	Reset configuration port - Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset It becomes functional when ARMv8-M Security Extensions are included When ARMv8-M Security Extensions are not included, this port will be ignored.
inittcmen[2]	Signal	Slave	Reset configuration port - TCM enable initialisation out of reset Bit[0] HIGH = ITCM is enabled Bit[1] HIGH = DTCM is enabled
intnum	Value	Master	Exception number of the current execution context (from IPSR[8:0]) When the processor is in Thread mode, INTNUM is 0 When the processor is in Handler mode, INTNUM is the exception number of the currently-executing exception.
irq[480]	Signal	Slave	This signal array delivers signals to the NVIC.
lockdcaic	Signal	Slave	Disable access to instruction cache direct cache access registers
lockdtgu	Signal	Slave	Disable writes to registers associated with the DTGU
lockitgu	Signal	Slave	Disable writes to registers associated with the ITGU
locknsmpu	Signal	Slave	Disable writes to the Non-Secure MPU_*_NS registers
locknsvtor	Signal	Slave	Cortex-M55-specific LOCKNSVTOR, LOCKSVTAIRCR, LOCKSMPU, LOCKNSMPU, LOCKSAU. Disable writes to VTOR_NS
lockpahb	Signal	Slave	P-AHB related ports Disable writes to PAHBSCR
locksau	Signal	Slave	Disable writes to the SAU_* registers
locksmpu	Signal	Slave	Disable writes to the Secure MPU_* registers
locksvtaircr	Signal	Slave	Disable writes to VTOR_S, AIRCR.PRIS, AIRCR.BFHFNMINS
locktcm	Signal	Slave	Disable writes to registers associated with the TCM
lockup	Signal	Master	Asserted when the processor is in lockup state.
niden	Signal	Slave	Non-invasive debug enable
nmi	Signal	Slave	Configure non maskable interrupt.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
prequest	PChannel	Slave	Low Power Interface
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pdbus_m	PVBus	Master	The core will generate bus requests on this port.
qrequest	PChannel	Slave	-

Name	Protocol	Type	Description
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
spiden	Signal	Slave	Secure invasive debug enable
spniden	Signal	Slave	Secure non-invasive debug enable
stcalib[2]	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.
wicenack	Signal	Master	Acknowledge signal for WICENREQ
wicenreq	Signal	Slave	Request for deep sleep to be WIC-based deep sleep.
wicsense[483]	Signal	Master	Indicates which input events can cause the WIC to generate the WAKEUP signal.

Parameters for ARM CortexM55CT

CDEMAPPEDONCP

Bit N specifies whether the instruction for coprocessor N (CP7:CP0) is redirected to the CDE module

Type

int

Default value

0xff

CDERTLID

Value of ID_AFR0.CDERTLID

Type

int

Default value

0x20

CFGBIGEND

Initialize processor to big endian mode

Type

bool

Default value

0x0

CFGDTCMSZ

Size of the data TCM. 0=No DTCM implemented. Otherwise=Size of DTCM=pow(2, CFGDTCMSZ - 1) KB. Minimum size is 4KB

Type

int

Default value

0x9

CFGITCMSZ

Size of the instruction TCM. 0=No ITCM implemented. Otherwise=Size of ITCM=pow(2, CFGITCMSZ - 1) KB. Minimum size is 4KB

Type

int

Default value

0x9

CFGMEMALIAS

Memory address alias bit for the ITCM, DTCM and P-AHB regions. 0=No alias, 1=Alias bit 24, 2=Alias bit 25, 4=Alias bit 26, 8=Alias bit 27, 16=Alias bit 28

Type

int

Default value

0x0

CFGNOCDECP

Bit N means external coprocessor N (CP7:CP0) disable for CDE coprocessor

Type

int

Default value

0x0

CFGPAHBSZ

Size of the P-AHB peripheral port memory region. 0=P-AHB disabled, 1=64MB, 2=128MB, 3=256MB, 4=512MB

Type

int

Default value

0x0

CPIF

Specifies whether the external coprocessor interface is included

Type

bool

Default value

0x1

CPNSPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Non-Secure state

Type

int

Default value

0xff

CPSPPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Secure state

Type

int

Default value

0xff

CTI

CTI (Cross Trigger Interface) included

Type

bool

Default value

0x0

CTI_irq0_pin

CTI interrupt request 0 pin

Type

int

Default value

0x4

CTI_irq1_pin

CTI interrupt request 1 pin

Type

int

Default value

0x5

DBGVLVL

0: Minimal debug; 1: 2 Watchpoints, 4 Breakpoint comparators; 2: 4 Watchpoints, 8 Breakpoint comparators; 3: 8 Watchpoints, 8 Breakpoint comparators

Type

int

Default value

0x2

DCACHESZ

Whether the D-cache is included and, if included, the size of it. Bit 0: 0=No D-cache included, 1=D-cache included. Bits [4:1]: 0x0=4KB D-cache, 0x1=8KB D-cache, 0x3=16KB D-cache, 0x7=32KB D-cache, 0xF=64KB D-cache

Type

int

Default value

0xf

DTGU

DTCM Security Gate Unit included

Type

bool

Default value

0x0

DTGUBLKSZ

DTCM gate unit block size. Size= $\text{pow}(2, \text{DTGUBLKSZ} + 5)$ bytes

Type

int

Default value

0x3

DTGUMAXBLKS

Maximum number of DTCM gate unit blocks. Number of blocks= $\text{pow}(2, \text{DTGUMAXBLKS})$

Type

int

Default value

0x0

ECOREVNUM

ECO Revision number

Type

int

Default value

0x0

ERRDEVID.NUM

RAS: Number of implemented error record indexes, 0 to 1.

Type

int

Default value

0x1

ETM

Support for ETM trace. false : No ETM trace included, true: ETM trace included

Type

bool

Default value

0x1

FPU

Set whether the model has VFP support

Type

bool

Default value

0x1

ICACHESZ

Whether the I-cache is included and, if included, the size of it. Bit 0: 0=No I-cache included, 1=I-cache included. Bits [4:1]: 0x0=4KB I-cache, 0x1=8KB I-cache, 0x3=16KB I-cache, 0x7=32KB I-cache, 0xF=64KB I-cache

Type

int

Default value

0xf

INITNSVTOR

Non-Secure vector-table offset at reset

Type

int

Default value

0x0

INITPAHBEN

The P-AHB enable state at reset

Type

bool

Default value

0x0

INITSVTOR

Secure vector-table offset at reset

Type

int

Default value

0x0

INITTCMEN

The TCM enable state at reset. Bit 0 corresponds to ITCM enable state, bit 1 corresponds to DTCM enable state

Type

int

Default value

0x3

IRQDIS0

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+0]

Type

int

Default value

0x0

IRQDIS1

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+32]

Type

int

Default value

0x0

IRQDIS10

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+320]

Type

int

Default value

0x0

IRQDIS11

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+352]

Type

int

Default value

0x0

IRQDIS12

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+384]

Type

int

Default value

0x0

IRQDIS13

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+416]

Type

int

Default value

0x0

IRQDIS14

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+448]

Type

int

Default value

0x0

IRQDIS2

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+64]

Type

int

Default value

0x0

IRQDIS3

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+96]

Type

int

Default value

0x0

IRQDIS4

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+128]

Type

int

Default value

0x0

IRQDIS5

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+160]

Type

int

Default value

0x0

IRQDIS6

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+192]

Type

int

Default value

0x0

IRQDIS7

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+224]

Type

int

Default value

0x0

IRQDIS8

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+256]

Type

int

Default value

0x0

IRQDIS9

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+288]

Type

int

Default value

0x0

IRQLVL

Number of bits of interrupt priority

Type

int

Default value

0x3

ITGU

ITCM Security Gate Unit included

Type

bool

Default value

0x0

ITGUBLKSZ

ITCM gate unit block size. Size=pow(2, ITGUBLKSZ + 5) bytes

Type

int

Default value

0x3

ITGUMAXBLKS

Maximum number of ITCM gate unit blocks. Number of blocks=pow(2, ITGUMAXBLKS)

Type

int

Default value

0x0

ITM

Level of instrumentation trace supported. false : No ITM trace included, true: ITM trace included

Type

bool

Default value

0x1

IWIC

Include support for Internal Wake-up Interrupt Controller

Type

bool

Default value

0x1

LOCKDTGU

Lock down of Data TGU registers write

Type

bool

Default value

0x0

LOCKITGU

Lock down of Instruction TGU registers write

Type

bool

Default value

0x0

LOCKTCM

Lock down of TCM registers write

Type

bool

Default value

0x0

LOCK_NS_MPU

Lock down of Non-Secure MPU registers write

Type

bool

Default value

0x0

LOCK_SAU

Lock down of SAU registers write

Type

bool

Default value

0x0

LOCK_S_MPU

Lock down of Secure MPU registers write

Type

bool

Default value

0x0

MPU_NS

Number of regions in the Non-Secure MPU. If Security Extensions are absent, this is the total number of MPU regions

Type

int

Default value

0x8

MPU_S

Number of regions in the Secure MPU. If Security Extensions are absent, this is ignored

Type

int

Default value

0x8

MVE

Set whether the model has MVE support. If FPU = 0: 0=MVE not included, 1=Integer subset of MVE included. If FPU = 1: 0=MVE not included, 1=Integer subset of MVE included, 2=Integer and half and single precision floating point MVE included

Type

int

Default value

0x1

NUMIRQ

Number of user interrupts

Type

int

Default value

0x20

SAU

Number of SAU regions (0 => no SAU)

Type

int

Default value

0x4

SECEXT

Whether the ARMv8-M Security Extensions are included

Type

bool

Default value

0x1

WICLINES

Number of lines supported by the WIC interface

Type

int

Default value

0x23

cde_impl_name

Name of the CDE implementation for this core (implementation contributed by MTI plugin).

Type

string

Default value**cpi_div**

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

delay_faultmask_update

Delay FAULTMASK update to context sync

Type

bool

Default value

0x0

delay_sysreg_update

Delay some system register updates (e.g. SHCSR) to context sync

Type

bool

Default value

0x0

ecc_on

Enable Error Correcting Code

Type

bool

Default value

0x0

has_cde

Enables Custom Datapath Extensions

Type

bool

Default value

0x0

has_core_dside_bus_gasket

STL gasket enabled

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

trace_style

MVE instruction trace style: Add 16 for [**-] beat trace. Add 32 for tracing IMPLIED LOB instructions. Add 64 to change opcode of implied BF to 0xBF00

Type

int

Default value

0x2

vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

3.5.60 ARM CortexM85CT

CortexM85CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-429: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters removed:

- BEATS_PER_TICK
- BF_is_nop
- ID_ISAR0.CmpBranch
- aircr_iesb_is_writable
- aircr_iesb_reset
- has_unpriviledged_debug
- num_pmu_counters
- ras_ERRFR0
- ras_cei_pin
- ras_cei_support
- ras_eri_pin
- ras_eri_support
- ras_error_record
- ras_fhi_pin
- ras_fhi_support

Iris and MTI instances for ARMCortexM85CT

This model has the following Iris instances:

Table 3-430: ARMCortexM85CT Iris instances

InstanceName	ComponentName
ARMCortexM85CT	ARM_Cortex-M85
ARMCortexM85CT.acp_mapper	PVBusMapper
ARMCortexM85CT.ext_bus	PVBusLogger
ARMCortexM85CT.ext_bus.mapper	PVBusMapper
ARMCortexM85CT.l1_incoherent_interconnect	PVCache
ARMCortexM85CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexM85CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexM85CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexM85CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexM85CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave

InstanceName	ComponentName
ARMCortexM85CT.11_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexM85CT.11dcache	PVCache
ARMCortexM85CT.11dcache.upstream[0]	PVBusSlave
ARMCortexM85CT.11icache	PVCache
ARMCortexM85CT.11icache.upstream[0]	PVBusSlave
ARMCortexM85CT.12_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-431: ARMCortexM85CT MTI instances

InstanceName	ComponentName
ARMCortexM85CT	ARM_Cortex-M85
ARMCortexM85CT.acp_mapper	PVBusMapper
ARMCortexM85CT.ext_bus	PVBusLogger
ARMCortexM85CT.ext_bus.mapper	PVBusMapper
ARMCortexM85CT.11_incoherent_interconnect	PVCache
ARMCortexM85CT.11_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[3]	PVBusSlave

InstanceName	ComponentName
ARMCortexM85CT.11_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexM85CT.11_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexM85CT.11dcache	PVCache
ARMCortexM85CT.11dcache.upstream[0]	PVBusSlave
ARMCortexM85CT.11icache	PVCache
ARMCortexM85CT.11icache.upstream[0]	PVBusSlave
ARMCortexM85CT.12_flusher	AsyncCacheFlushUnit

ARMCortexM85CT contains the following CADI targets:

- ARM_Cortex-M85

About ARMCortexM85CT

This model supports the M-Profile Vector Extension (MVE) and the Custom Datapath Extension (CDE). For more information about CDE support in Fast Models, see [4.7 CDE](#) on page 4148.

Ports for ARMCortexM85CT

Table 3-432: Ports

Name	Protocol	Type	Description
ahbd	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
ahbp_m	PVBus	Master	The core will generate Vendor System data accesses on this port.
ahbs	PVBus	Slave	Slave AHB - External master (e.g. DMA) can write to TCMs (whether or not enabled in xTCMCR)
auxfault	Value	Slave	This is wired to the Auxiliary Fault Status Register.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions. Referred to in some documents as 'CLKIN'.
coproc_bus	CoprocBusProtocol	Slave	Co-Processor Interface
core_dside_bus_gasket_in	PVBus	Slave	-
core_dside_bus_gasket_out	PVBus	Master	-
cpuwait	Signal	Slave	Stall the CPU out of reset
currns	Signal	Master	Current Security state of the processor
currpri	Value	Master	Current execution priority.
dbgen	Signal	Slave	Invasive debug enable
dbgrestart	Signal	Slave	Request for synchronised exit from halt mode

Name	Protocol	Type	Description
dbgrestarted	Signal	Master	Handshakes with DBGRESTART
edbgrq	Signal	Slave	External request to enter halt mode
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
fpxxc	Value	Master	Port that sends the value of the FPxxC exception flags (FPIXC, FPIDC, FPOFC, FPUFC, FPDZC, FPIOC). Each flag is present in the same position as in the FPSCR register (e.g. fpxxc[0] = FPIOC).
halted	Signal	Master	Indicates that the processor is in halt mode
idau	PVBus	Master	The core will generate IDAU requests on this port.
idau_invalidate_region	Value_64	Slave	64 bit number to invalid IDAU memory range (start_address<<32 end_address)
initnsvtor	Value	Slave	Reset configuration port - Non-Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset This port remains functional no matter ARMv8-M Security Extensions are included or not When ARMv8-M Security Extensions are not included, all exceptions will use NS vector base address given by this port.
initpahben	Signal	Slave	Enable P-AHB on the next reset
initsvtor	Value	Slave	Reset configuration port - Secure Vector table offset (VTOR.TBLOFF[31:7]) out of reset It becomes functional when ARMv8-M Security Extensions are included When ARMv8-M Security Extensions are not included, this port will be ignored.
inittcmem[2]	Signal	Slave	Reset configuration port - TCM enable initialisation out of reset Bit[0] HIGH = ITCM is enabled Bit[1] HIGH = DTCM is enabled
intnum	Value	Master	Exception number of the current execution context (from IPSR[8:0]) When the processor is in Thread mode, INTNUM is 0 When the processor is in Handler mode, INTNUM is the exception number of the currently-executing exception.
irq[480]	Signal	Slave	This signal array delivers signals to the NVIC.
lockdcaic	Signal	Slave	Disable access to instruction cache direct cache access registers
lockdtgu	Signal	Slave	Disable writes to registers associated with the DTGU
lockitgu	Signal	Slave	Disable writes to registers associated with the ITGU
locknsmpu	Signal	Slave	Disable writes to the Non-Secure MPU_*_NS registers
locknsvtor	Signal	Slave	Cortex-M85 specific LOCKNSVTOR, LOCKSVTAIRCR, LOCKSMPU, LOCKNSMPU, LOCKSAU. Disable writes to VTOR_NS
lockpahb	Signal	Slave	P-AHB related ports Disable writes to PAHBRCR
locksau	Signal	Slave	Disable writes to the SAU_* registers
locksmpu	Signal	Slave	Disable writes to the Secure MPU_* registers
locksvtaircr	Signal	Slave	Disable writes to VTOR_S, AIRCR.PRIS, AIRCR.BFHFNMINS
locktcm	Signal	Slave	Disable writes to registers associated with the TCM
lockup	Signal	Master	Asserted when the processor is in lockup state.
niden	Signal	Slave	Non-invasive debug enable
nmi	Signal	Slave	Configure non maskable interrupt.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
prequest	PChannel	Slave	Low Power Interface

Name	Protocol	Type	Description
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbus_m	PVBus	Master	The core will generate bus requests on this port.
qrequest	PChannel	Slave	-
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
spiden	Signal	Slave	Secure invasive debug enable
spniden	Signal	Slave	Secure non-invasive debug enable
stcalib[2]	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.
wicenack	Signal	Master	Acknowledge signal for WICENREQ
wicenreq	Signal	Slave	Request for deep sleep to be WIC-based deep sleep.
wicsense[483]	Signal	Master	Indicates which input events can cause the WIC to generate the WAKEUP signal.

Parameters for ARM CortexM85CT

CDEMAPPEDONCP

Bit N specifies whether the instruction for coprocessor N (CP7:CP0) is redirected to the CDE module

Type

int

Default value

0xff

CDERTLID

Value of ID_AFR0.CDERTLID

Type

int

Default value

0x20

CFGBIGEND

Initialize processor to big endian mode

Type

bool

Default value

0x0

CFGCPUINST

CPU instance number. This is part of the TCM base address, in bits 25:24.

Type

int

Default value

0x0

CFGDTCMSZ

Size of the data TCM. 0=No DTCM implemented. Otherwise=Size of DTCM=pow(2, CFGDTCMSZ - 1) KB. Minimum size is 4KB

Type

int

Default value

0x9

CFGITCMSZ

Size of the instruction TCM. 0=No ITCM implemented. Otherwise=Size of ITCM=pow(2, CFGITCMSZ - 1) KB. Minimum size is 4KB

Type

int

Default value

0x9

CFGMEMALIAS

Memory address alias bit for the ITCM, DTCM and P-AHB regions. 0=No alias, 1=Alias bit 24, 2=Alias bit 25, 4=Alias bit 26, 8=Alias bit 27, 16=Alias bit 28

Type

int

Default value

0x0

CFGNOCDECP

Bit N means external coprocessor N (CP7:CP0) disable for CDE coprocessor

Type

int

Default value

0x0

CFGPAACBTI

Enables support for the Pointer Authentication and Branch Target Identification (PACBTI) Extension. FALSE: Disabled, TRUE: PAC implemented using the QARMA3 algorithm with BTI

Type

bool

Default value

0x0

CFGPAHBSZ

Size of the P-AHB peripheral port memory region. 0=P-AHB disabled, 1=64MB, 2=128MB, 3=256MB, 4=512MB

Type

int

Default value

0x0

CPIF

Specifies whether the external coprocessor interface is included

Type

bool

Default value

0x1

CPNSPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Non-Secure state

Type

int

Default value

0xff

CPSPRESENT

Bit N means external coprocessor N (CP7:CP0) is accessible in Secure state

Type

int

Default value

0xff

CTI

CTI (Cross Trigger Interface) included

Type

bool

Default value

0x0

CTI_irq0_pin

CTI interrupt request 0 pin

Type

int

Default value

0x4

CTI_irq1_pin

CTI interrupt request 1 pin

Type

int

Default value

0x5

DBGLVL

1: 4 Watchpoints, 4 Breakpoint comparators; 2: 8 Watchpoints, 8 Breakpoint comparators

Type

int

Default value

0x2

DCACHESZ

Whether the D-cache is included and, if included, the size of it. Bit 0: 0=No D-cache included, 1=D-cache included. Bits [4:1]: 0x0=4KB D-cache, 0x1=8KB D-cache, 0x3=16KB D-cache, 0x7=32KB D-cache, 0xF=64KB D-cache

Type

int

Default value

0xf

DTGU

DTCM Security Gate Unit included

Type

bool

Default value

0x0

DTGUBLKSZ

DTCM gate unit block size. Size=pow(2, DTGUBLKSZ + 5) bytes

Type

int

Default value

0x3

DTGUMAXBLKS

Maximum number of DTCM gate unit blocks. Number of blocks=pow(2, DTGUMAXBLKS)

Type

int

Default value

0x0

ECOREVNUM

ECO Revision number

Type

int

Default value

0x0

ERRDEVID.NUM

RAS: Number of implemented error record indexes, 0 to 1.

Type

int

Default value

0x1

ETM

Support for ETM trace. false : No ETM trace included, true: ETM trace included

Type

bool

Default value

0x1

FPU

Set whether the model has VFP support

Type

bool

Default value

0x1

ICACHESZ

Whether the I-cache is included and, if included, the size of it. Bit 0: 0=No I-cache included, 1=I-cache included. Bits [4:1]: 0x0=4KB I-cache, 0x1=8KB I-cache, 0x3=16KB I-cache, 0x7=32KB I-cache, 0xF=64KB I-cache

Type

int

Default value

0xf

INITNSVTOR

Non-Secure vector-table offset at reset

Type

int

Default value

0x0

INITPAHBEN

The P-AHB enable state at reset

Type

bool

Default value

0x0

INITSVTOR

Secure vector-table offset at reset

Type

int

Default value

0x0

INITTCMEN

The TCM enable state at reset. Bit 0 corresponds to ITCM enable state, bit 1 corresponds to DTCM enable state

Type

int

Default value

0x3

IRQDIS0

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+0]

Type

int

Default value

0x0

IRQDIS1

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+32]

Type

int

Default value

0x0

IRQDIS10

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+320]

Type

int

Default value

0x0

IRQDIS11

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+352]

Type

int

Default value

0x0

IRQDIS12

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+384]

Type

int

Default value

0x0

IRQDIS13

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+416]

Type

int

Default value

0x0

IRQDIS14

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+448]

Type

int

Default value

0x0

IRQDIS2

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+64]

Type

int

Default value

0x0

IRQDIS3

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+96]

Type

int

Default value

0x0

IRQDIS4

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+128]

Type

int

Default value

0x0

IRQDIS5

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+160]

Type

int

Default value

0x0

IRQDIS6

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+192]

Type

int

Default value

0x0

IRQDIS7

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+224]

Type

int

Default value

0x0

IRQDIS8

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+256]

Type

int

Default value

0x0

IRQDIS9

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n+288]

Type

int

Default value

0x0

IRQLVL

Number of bits of interrupt priority

Type

int

Default value

0x3

ITGU

ITCM Security Gate Unit included

Type

bool

Default value

0x0

ITGUBLKSZ

ITCM gate unit block size. Size=pow(2, ITGUBLKSZ + 5) bytes

Type

int

Default value

0x3

ITGUMAXBLKS

Maximum number of ITCM gate unit blocks. Number of blocks= $\text{pow}(2, \text{ITGUMAXBLKS})$

Type

int

Default value

0x0

ITM

Level of instrumentation trace supported. false : No ITM trace included, true: ITM trace included

Type

bool

Default value

0x1

IWIC

Include support for Internal Wake-up Interrupt Controller

Type

bool

Default value

0x1

LOCKDTGU

Lock down of Data TGU registers write

Type

bool

Default value

0x0

LOCKITGU

Lock down of Instruction TGU registers write

Type

bool

Default value

0x0

LOCKTCM

Lock down of TCM registers write

Type

bool

Default value

0x0

LOCK_NS_MPU

Lock down of Non-Secure MPU registers write

Type

bool

Default value

0x0

LOCK_SAU

Lock down of SAU registers write

Type

bool

Default value

0x0

LOCK_S_MPU

Lock down of Secure MPU registers write

Type

bool

Default value

0x0

MPU_NS

Number of regions in the Non-Secure MPU. If Security Extensions are absent, this is the total number of MPU regions

Type

int

Default value

0x8

MPU_S

Number of regions in the Secure MPU. If Security Extensions are absent, this is ignored

Type

int

Default value

0x8

MVE

Set whether the model has MVE support. If FPU = 0: 0=MVE not included, 1=Integer subset of MVE included. If FPU = 1: 0=MVE not included, 1=Integer subset of MVE included, 2=Integer and half and single precision floating point MVE included

Type

int

Default value

0x1

NUMIRQ

Number of user interrupts

Type

int

Default value

0x20

SAU

Number of SAU regions (0 => no SAU)

Type

int

Default value

0x4

SECEXT

Whether the ARMv8-M Security Extensions are included

Type

bool

Default value

0x1

WICLINES

Number of lines supported by the WIC interface

Type

int

Default value

0x23

cde_impl_name

Name of the CDE implementation for this core (implementation contributed by MTI plugin).

Type

string

Default value**cpi_div**

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

delay_faultmask_update

Delay FAULTMASK update to context sync

Type

bool

Default value

0x0

delay_sysreg_update

Delay some system register updates (e.g. SHCSR) to context sync

Type

bool

Default value

0x0

ecc_on

Enable Error Correcting Code

Type

bool

Default value

0x0

has_cde

Enables Custom Datapath Extensions

Type

bool

Default value

0x0

has_core_dside_bus_gasket

STL gasket enabled

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CPUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

trace_style

MVE instruction trace style: Add 16 for [**--] beat trace. Add 32 for tracing IMPLIED LOB instructions. Add 64 to change opcode of implied BF to 0xBF00

Type

int

Default value

0x2

vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

3.5.61 ARMCortexR4CT

ARMCortexR4CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-433: IP revisions support

Revision	Quality level
r1p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexR4CT

This model has the following Iris instances:

Table 3-434: ARMCortexR4CT Iris instances

InstanceName	ComponentName
ARMCortexR4CT	ARM_Cortex-R4
ARMCortexR4CT.acp_mapper	PVBusMapper
ARMCortexR4CT.cpu0.l1dcache	PVCache
ARMCortexR4CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR4CT.cpu0.l1icache	PVCache
ARMCortexR4CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexR4CT.ext_bus	PVBusLogger
ARMCortexR4CT.ext_bus.mapper	PVBusMapper
ARMCortexR4CT.l1_incoherent_interconnect	PVCache
ARMCortexR4CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave

InstanceName	ComponentName
ARMCortexR4CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexR4CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-435: ARMCortexR4CT MTI instances

InstanceName	ComponentName
ARMCortexR4CT	ARM_Cortex-R4
ARMCortexR4CT.acp_mapper	PVBusMapper
ARMCortexR4CT.cpu0.l1dcache	PVCache
ARMCortexR4CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR4CT.cpu0.l1icache	PVCache
ARMCortexR4CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexR4CT.ext_bus	PVBusLogger
ARMCortexR4CT.ext_bus.mapper	PVBusMapper
ARMCortexR4CT.l1_incoherent_interconnect	PVCache
ARMCortexR4CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexR4CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexR4CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexR4CT contains the following CADI targets:

- ARM_Cortex-R4
- PVCache

About ARMCortexR4CT

- The model implements the `cfgie` port, although it is optional in hardware.
- `pvtbus_s` is the slave port to access the TCM RAM. Bits [3:0] of the user flags in the transaction are used to select the TCM:
 - 1 selects the ATCM.
 - 2 selects the BTCM.
 - Any other value is reserved.
- When you use the `semihosting-cmd_line` parameter to set the command line that is available to semihosting SVC calls, the value of `argv[0]` is set to the first command-line argument, not to the name of an image.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- There is a single memory port combining instruction, data, DMA and peripheral access.
- ECC and parity schemes are not supported (although the registers might be present).
- The dual core redundancy configuration is not supported.
- TCMs are modeled internally and the model does not support external TCMs or the ports associated with them.
- The hardware refers to the TCMs as “A” and “B”. The model refers to these as “i” and “d”.
- The RTL permits two data TCMs, B0 and B1, to be configured for extra bandwidth. These are not modeled.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.

Vectored Interrupt Controller (VIC) ports

The ARMCortexR4CT and ARMCortexR5xnCT models implement a simplified model of the Vectored Interrupt Controller (VIC) port.

The protocol consists of two ports:

- The `vic_addr` port signals the vectored interrupt address from the external VIC.
- The `vic_ack` port signals the VIC that an interrupt has been detected and is being serviced.

The expected interrupt sequence is:

1. The software enables the VIC interface by setting the VE bit in the CP15 control register and setting up suitable interrupt routines.
2. The VIC asserts IRQ.

3. Some time later, the processor detects and responds to the IRQ by asserting `vic_ack`.
4. The VIC writes the vector address to the processor using `vic_addr`.
5. The processor de-asserts `vic_ack`.
6. The processor transfers control to the vectored address returned from the VIC.

The interaction between the processor and the VIC is untimed after the processor acknowledges the interrupt, so certain interrupt sequences cannot occur in the code translation processor models.

Ports for ARM CortexR4CT

Table 3-436: Ports

Name	Protocol	Type	Description
<code>cfgend0</code>	Signal	Slave	Configure BE8 mode after a reset.
<code>cfgie</code>	Signal	Slave	Configure big endian instruction format after a reset.
<code>cfgnmfi</code>	Signal	Slave	Configure FIQs as non-maskable after a reset.
<code>cfgte</code>	Signal	Slave	Configure exceptions to be taken in thumb mode after a reset.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>cpuhalt</code>	Signal	Slave	Raising this signal will put the core into halt mode.
<code>fiq</code>	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
<code>initramd</code>	Signal	Slave	Configure DTCM enabled after a reset.
<code>initrami</code>	Signal	Slave	Configure ITCM enabled after a reset.
<code>irq</code>	Signal	Slave	This signal drives the CPU's interrupt handling.
<code>loczrama</code>	Signal	Slave	Location of ATCM at reset.
<code>pmuirq</code>	Signal	Master	Interrupt signal from performance monitoring unit.
<code>pvbus_m</code>	PVBus	Master	The core will generate bus requests on this port.
<code>pvbus_s</code>	PVBus	Slave	Slave access to TCMs.
<code>reset</code>	Signal	Slave	Raising this signal will put the core into reset mode.
<code>standbywfi</code>	Signal	Master	Signal from the core that it is waiting in standby for an interrupt.
<code>ticks</code>	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.
<code>vic_ack</code>	Signal	Master	Vic acknowledge port to primary VIC.
<code>vic_addr</code>	Value	Slave	Vic address port from primary VIC.
<code>vinithi</code>	Signal	Slave	Configure high vectors after a reset.

Parameters for ARM CortexR4CT

CFGEND0

Initialize to BE8 endianness

Type

bool

Default value

0x0

CFGIE

Set the reset value of the instruction endian bit

Type

bool

Default value

0x0

CFGNMFI

Enable nonmaskable FIQ interrupts on startup

Type

bool

Default value

0x0

CFGTE

Initialize to take exceptions in T32 state. Model starts in T32 state

Type

bool

Default value

0x0

INITRAMD

Set or reset the INITRAMD signal

Type

bool

Default value

0x0

INITRAMI

Set or reset the INITRAMI signal

Type

bool

Default value

0x0

LOCZRAMI

Set or reset the LOCZRAMI signal

Type

bool

Default value

0x0

NUM_MPU_REGION

Number of MPU regions

Type

int

Default value

0x8

VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

dcache-size

Set D-cache size in bytes

Type

int

Default value

0x10000

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dtcm0_base

Base address of DTCM at startup

Type

int

Default value

0x800000

dtcm0_size

Size of DTCM in KB

Type

int

Default value

0x8

icache-size

Set I-cache size in bytes

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

implements_vfp

Set whether the model has been built with VFP support

Type

bool

Default value

0x1

itcm0_base

Base address of ITCM at startup

Type

int

Default value

0x0

itcm0_size

Size of ITCM in KB

Type

int

Default value

0x8

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

scheduler_mode

Control the interleaving of instructions in this processor. 0, default long quantum. 1, low latency mode, short quantum and signal checking. 2, lock-breaking mode, long quantum with additional context switches near load-exclusive instructions. WARNING: This parameter is intended for validation purposes and may result in unwanted behaviour if altered!

Type

int

Default value

0x0

semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

3.5.62 ARM CortexR5x1CT

ARM CortexR5x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-437: IP revisions support

Revision	Quality level
r1p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARM CortexR5x1CT

This model has the following Iris instances:

Table 3-438: ARM CortexR5x1CT Iris instances

InstanceName	ComponentName
ARM CortexR5x1CT	Cluster_ARM_Cortex-R5
ARM CortexR5x1CT.acp_mapper	PVBusMapper
ARM CortexR5x1CT.cpu0	ARM_Cortex-R5
ARM CortexR5x1CT.cpu0.l1dcache	PVCache
ARM CortexR5x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM CortexR5x1CT.cpu0.l1icache	PVCache
ARM CortexR5x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM CortexR5x1CT.ext_bus	PVBusLogger
ARM CortexR5x1CT.ext_bus.mapper	PVBusMapper
ARM CortexR5x1CT.l1_incoherent_interconnect	PVCache
ARM CortexR5x1CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARM CortexR5x1CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARM CortexR5x1CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARM CortexR5x1CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARM CortexR5x1CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARM CortexR5x1CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARM CortexR5x1CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave

InstanceName	ComponentName
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexR5x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-439: ARMCortexR5x1CT MTI instances

InstanceName	ComponentName
ARMCortexR5x1CT.acp_mapper	PVBusMapper
ARMCortexR5x1CT.cpu0	ARM_Cortex-R5
ARMCortexR5x1CT.cpu0.l1dcache	PVCache
ARMCortexR5x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR5x1CT.cpu0.l1icache	PVCache
ARMCortexR5x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexR5x1CT.ext_bus	PVBusLogger
ARMCortexR5x1CT.ext_bus.mapper	PVBusMapper
ARMCortexR5x1CT.l1_incoherent_interconnect	PVCache
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave

InstanceName	ComponentName
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexR5x1CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexR5x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexR5x1CT contains the following CADI targets:

- ARM_Cortex-R5
- Cluster_ARM_Cortex-R5

About ARMCortexR5x1CT

- An ARMCortexR5x2CT component also exists.
- The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0 or 1).
- The allowed values for the `LOCK_STEP` parameter are:

0	Disable. Set for two independent cores.
1	Lock Step. Appears to the system as two cores but is internally modeled as a single core.
3	Split Lock. Appears to the system as two cores but can be statically configured from reset either as two independent cores or two locked cores. For the model, these are equivalent to Disable and Lock Step, respectively, except for the value of build options registers. The model does not support dynamically splitting and locking the cluster.
- `pvbus_s` is the slave port to access the TCM RAM of CPU *n*. Bits [3:0] of the user flags in the transaction are used to select the TCM:
 - 1 selects the ATCM of CPU 0.
 - 2 selects the BTCM of CPU 0.
 - 3 selects the ATCM of CPU 1.
 - 4 selects the BTCM of CPU 1.
 - Any other value is reserved.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- The RR bit in the SCTL is ignored.
- The Low Latency Peripheral Port is not modeled.
- The model only has a single bus master port combining instruction, data, DMA and peripheral accesses. The CP15 control registers associated with peripheral buses preserve values but do not have any other effect.
- The model only supports static split lock and not dynamic split lock. Contact Arm for details.

- TCMs are modeled internally and the model does not support external TCMs or the ports associated with them.
- The model cannot experience an ECC error and does not support fault injection into the system, so Arm does not provide the ability to set error schemes for the caches or TCMs. Contact Arm if you require a particular value in the Build Options registers.

Vectored Interrupt Controller (VIC) ports

The ARMCortexR4CT and ARMCortexR5xnCT models implement a simplified model of the Vectored Interrupt Controller (VIC) port.

The protocol consists of two ports:

- The `vic_addr` port signals the vectored interrupt address from the external VIC.
- The `vic_ack` port signals the VIC that an interrupt has been detected and is being serviced.

The expected interrupt sequence is:

1. The software enables the VIC interface by setting the VE bit in the CP15 control register and setting up suitable interrupt routines.
2. The VIC asserts IRQ.
3. Some time later, the processor detects and responds to the IRQ by asserting `vic_ack`.
4. The VIC writes the vector address to the processor using `vic_addr`.
5. The processor de-asserts `vic_ack`.
6. The processor transfers control to the vectored address returned from the VIC.

The interaction between the processor and the VIC is untimed after the processor acknowledges the interrupt, so certain interrupt sequences cannot occur in the code translation processor models.

Ports for ARMCortexR5x1CT

Table 3-440: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	ACP slave port.
<code>cfgatcmsz[2]</code>	Value	Slave	ATCM size.
<code>cfgbtcmsz[2]</code>	Value	Slave	BTCLM Size.
<code>cfgend[2]</code>	Signal	Slave	This signal is for EE bit initialisation. This is CFGEE in RTL but <code>cfgend</code> here fastsim consistency reasons.
<code>cfgnmfi[2]</code>	Signal	Slave	Controls non-maskable Fast Interrupts.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>cpuhalt[2]</code>	Signal	Slave	Raising this signal will put the core into halt mode.
<code>event[2]</code>	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
<code>fiq[2]</code>	Signal	Slave	This signal drives the CPU's fast-interrupt handling.
<code>groupid</code>	Value	Slave	Group ID used for MPIDR.

Name	Protocol	Type	Description
initrama[2]	Signal	Slave	If ATCM is enabled at reset.
initramb[2]	Signal	Slave	If BTCM is enabled at reset.
irq[2]	Signal	Slave	This signal drives the CPU's interrupt handling.
loczrama[2]	Signal	Slave	Location of ATCM at reset.
pmuirq[2]	Signal	Master	Interrupt signal from performance monitoring unit.
pvbus_m	PVBus	Master	The core will generate bus requests on this port.
pvbus_s[1]	PVBus	Slave	tcm slave port.
reset[2]	Signal	Slave	Raising this signal will put the core into reset mode.
standbywfe[2]	Signal	Master	This signal indicate if a core is in wfe state RTL calls this WFEPIPESTOPPED.
standbywfi[2]	Signal	Master	This signal indicates if a core is in WFI state RTL uses WFIPIPESTOPPED.
teinit[2]	Signal	Slave	Default exception handling state.
ticks[2]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vic_ack[2]	Signal	Master	Vic acknowledge port to primary VIC.
vic_addr[2]	Value	Slave	Vic address port from primary VIC.
vinithi[2]	Signal	Slave	This signal controls of the location of the exception vectors at reset.

Parameters for ARM Cortex R5x1CT

GROUP_ID

Value read in GROUP ID register field, bits[15:8] of the MPIDR

Type

int

Default value

0x0

INST_ENDIAN

Controls whether the model supports the instruction endianness bit

Type

bool

Default value

0x1

LOCK_STEP

Affects dual-processor configurations only, and ignored by single-processor configurations

Type

int

Default value

0x0

MICRO_SCU

Controls whether the effects of the MicroSCU are modeled

Type

bool

Default value

0x1

NUM_BREAKPOINTS

Controls with how many breakpoint pairs the model has been configured. This only affects the build options registers, because debug is not modeled

Type

int

Default value

0x3

NUM_MPU_REGION

Sets the number of MPU regions

Type

int

Default value

0xc

NUM_WATCHPOINTS

Controls with how many watchpoint pairs the model has been configured. This only affects the build options registers, because debug is not modeled

Type

int

Default value

0x2

SLSPLIT

Sets whether the model starts in split mode or locked mode

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGATCMSZ

Sets the size of the ATCM

Type

int

Default value

0xe

cpuX.CFGBTCMSZ

Sets the size of the BTCM

Type

int

Default value

0xe

cpuX.CFGEND

Initialize to BE8 endianness

Type

bool

Default value

0x0

cpuX.CFGIE

Set the reset value of the instruction endian bit

Type

bool

Default value

0x0

cpuX.CFGNMFI

Enable nonmaskable FIQ interrupts on startup

Type

bool

Default value

0x0

cpuX.DP_FLOAT

Sets whether double-precision instructions are available

Type

bool

Default value

0x1

cpuX.INITRAMA

Initialize with TCMA enabled

Type

bool

Default value

0x0

cpuX.INITRAMB

Initialize with TCMB enabled

Type

bool

Default value

0x0

cpuX.LOCZRAMA

Initialize with LOCZRAMA set to 1

Type

bool

Default value

0x0

cpuX.TEINIT

T32 exception enable. The default has exceptions including reset handled in A32 state

Type

bool

Default value

0x0

cpuX.VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpuX.atcm_base

Model-specific. Sets the base address of the ATCM (forced to 0 if LOCZRAMA is 1)

Type

int

Default value

0x40000000

cpuX.btcn_base

Model-specific. Sets the base address of the BTCM (forced to 0 if LOCZRAMA is 0)

Type

int

Default value

0x0

cpuX.dcache-size

Set D-cache size in bytes

Type

int

Default value

0x10000

cpuX.icache-size

Set I-cache size in bytes

Type

int

Default value

0x10000

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

cpuX.semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value

cpuX.semihosting-enable

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

cpuX.semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

cpuX.vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether model has VFP support

Type

bool

Default value

0x1

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

scheduler_mode

Control the interleaving of instructions in this processor. 0, default long quantum. 1, low latency mode, short quantum and signal checking. 2, lock-breaking mode, long quantum with

additional context switches near load-exclusive instructions. WARNING: This parameter is intended for validation purposes and may result in unwanted behaviour if altered!

Type

int

Default value

0x0

3.5.63 ARM Cortex R7x1CT

ARM Cortex R7x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-441: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARM Cortex R7x1CT

This model has the following Iris instances:

Table 3-442: ARM Cortex R7x1CT Iris instances

InstanceName	ComponentName
ARM Cortex R7x1CT	Cluster_ARM_Cortex-R7
ARM Cortex R7x1CT.acp_mapper	PVBusMapper
ARM Cortex R7x1CT.cpu0	ARM_Cortex-R7
ARM Cortex R7x1CT.cpu0.l1dcache	PVCache
ARM Cortex R7x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM Cortex R7x1CT.cpu0.l1icache	PVCache
ARM Cortex R7x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM Cortex R7x1CT.ext_bus	PVBusLogger
ARM Cortex R7x1CT.ext_bus.mapper	PVBusMapper
ARM Cortex R7x1CT.l1_incoherent_interconnect	PVCache
ARM Cortex R7x1CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARM Cortex R7x1CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARM Cortex R7x1CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARM Cortex R7x1CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARM Cortex R7x1CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARM Cortex R7x1CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARM Cortex R7x1CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARM Cortex R7x1CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave

InstanceName	ComponentName
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexR7x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-443: ARMCortexR7x1CT MTI instances

InstanceName	ComponentName
ARMCortexR7x1CT.acp_mapper	PVBusMapper
ARMCortexR7x1CT.cpu0	ARM_Cortex-R7
ARMCortexR7x1CT.cpu0.l1dcache	PVCache
ARMCortexR7x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR7x1CT.cpu0.l1licache	PVCache
ARMCortexR7x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexR7x1CT.ext_bus	PVBusLogger
ARMCortexR7x1CT.ext_bus.mapper	PVBusMapper
ARMCortexR7x1CT.l1_incoherent_interconnect	PVCache
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave

InstanceName	ComponentName
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexR7x1CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexR7x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexR7x1CT contains the following CADI targets:

- ARM_Cortex-R7
- Cluster_ARM_Cortex-R7
- PVCache

About ARMCortexR7x1CT

- An ARMCortexR7x2CT component also exists.
- The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0 or 1).
- `pvbuss` is the slave port to access the TCM RAM of CPU *n*. Bits [3:0] of the user flags in the transaction are used to select the TCM:
 - 0 selects the ITCM of CPU 0.
 - 1 selects the DTCM of CPU 0.
 - 2 selects the ITCM of CPU 1.
 - 3 selects the DTCM of CPU 1.
 - Any other value is reserved.
- When you use the `semihosting-cmd_line` parameter to set the command line that is available to semihosting SVC calls, the value of `argv[0]` is set to the first command-line argument, not to the name of an image.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- This component does not implement address filtering within the SCU. The enable bit for this feature is ignored.
- The GIC does not respect the CFGSDISABLE signal. This leads to some registers being accessible when they must not be.
- The SCU enable bit is ignored. The SCU is always enabled.
- The SCU ignores the invalidate all register.
- The Broadcast TLB or cache operations in this model do not cause other cores in the cluster that are asleep because of WFI to wake up.
- The RR bit in the SCTLR is ignored.

- The Power Control Register in the system control coprocessor is implemented but writing to it does not change the behavior of the model.
- The model cannot be configured with a 128-entry TLB.
- When modeling the SCU, coherency operations are represented by a memory write followed by a read to refill from memory, rather than using cache-to-cache transfers.
- TCMs are modeled internally and the model does not support external TCMs or the ports associated with them.
- This component implements L1 cache as architecturally defined, but does not implement L2 cache. If you require an L2 cache you can add a PL310 Level 2 Cache Controller component.
- ECC and parity schemes are hardware-specific so are not supported.

Ports for ARMCortexR7x1CT

Table 3-444: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
cfgend[1]	Signal	Slave	This signal is for EE bit initialisation.
cfgnmfi[1]	Signal	Slave	This signal disables FIQ mask in CPSR.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[1]	Signal	Slave	Legacy FIQ request input line.
fiqout[1]	Signal	Master	Output of individual processor nFIQ from the interrupt controller.
fpuflags[1]	ValueState	Master	Floating-Point Unit output flags.
halt[1]	Signal	Slave	Raising this signal will put the core into halt mode. Equivalent to the hardware nCPUHALT[N:0] signal.
initram[1]	Signal	Slave	This signal enables the processor to boot from the instruction TCM.
ints[480]	Signal	Slave	Interrupt distributor interrupt lines.
irq[1]	Signal	Slave	Legacy IRQ request input line.
irqout[1]	Signal	Master	Output of individual processor nIRQ from the interrupt controller.
mfilteren	Signal	Slave	This signal enables filtering of address ranges between master bus ports.
mfilterend	Value	Slave	This port sets end of region mapped to pvbus_m1.
mfilterstart	Value	Slave	This port sets start of region mapped to pvbus_m1.
periphbase	Value	Slave	This port sets the base address of private peripheral region.
periphclk_in	ClockSignal	Slave	The timer and the watchdog take need a clk that is scaled down at least by factor of two.
periphreset	Signal	Slave	This signal resets timer and interrupt controller.
pfilterend	Value	Slave	This port sets end of region mapped to pvbus_mp.
pfilterstart	Value	Slave	This port sets start of region mapped to pvbus_mp.
pmuirq[2]	Signal	Master	Interrupt signal from performance monitoring unit.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.

Name	Protocol	Type	Description
pvbuss_m1	PVBus	Master	The core will generate bus requests on this port.
pvbuss_mp	PVBus	Master	The core will generate bus requests on this port.
pvbuss_s	PVBus	Slave	tcm slave port
reset[1]	Signal	Slave	Raising this signal will put the core into reset mode.
scureset	Signal	Slave	This signal resets SCU.
smpnamp[1]	Signal	Master	This signals AMP or SMP mode for each Cortex-R7 processor.
standbywfe[1]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[1]	Signal	Master	This signal indicates if a core is in WFI state.
teinit[1]	Signal	Slave	This signal provides default exception handling state.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vinithi[1]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
wdreset[1]	Signal	Slave	This signal resets individual watchdog.
wdresetreq[1]	Signal	Master	CPU watchdog reset requests.

Parameters for ARMCortexR7x1CT

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

LOCK_STEP

Affects dual-processor configurations only, and ignored by single-processor configurations. 0 - Disable. Set for two independent processors. 1 - Lock Step. Appears to the system as two processors but is internally modeled as a single processor. 3 - Split Lock. Appears to the system as two processors but can be statically configured from reset either as two independent processors or two locked processors. For the model, these are equivalent to Disable and Lock Step, respectively, except for the value of build options registers. The model does not support dynamically splitting and locking the processor.

Type

int

Default value

0x0

MFILTEREN

Enables filtering of address ranges

Type

bool

Default value

0x0

MFILTEREND

Specifies the end address for address filtering

Type

int

Default value

0x0

MFILTERSTART

Specifies the start address for address filtering

Type

int

Default value

0x0

NUM_MPU_REGION

Sets the number of MPU regions

Type

int

Default value

0xc

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0xae000000

PFILTEREND

Specifies the end address for peripheral port address filtering

Type

int

Default value

0x0

PFILTERSTART

Specifies the start address for peripheral port address filtering

Type

int

Default value

0xffff00000

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Initialize to BE8 endianness

Type

bool

Default value

0x0

cpuX.CFGNMFI

Enable nonmaskable FIQ interrupts on startup

Type

bool

Default value

0x0

cpuX.DP_FLOAT

Sets whether double-precision instructions are available

Type

bool

Default value

0x1

cpuX.INITRAM

Enable the processor to boot from the instruction TCM

Type

bool

Default value

0x0

cpuX.POWERCTLI

Default power control state for processor

Type

int

Default value

0x0

cpuX.SMPnAMP

Set whether the processor is part of a coherent domain

Type

bool

Default value

0x0

cpuX.TEINIT

T32 exception enable. The default has exceptions including reset handled in A32 state

Type

bool

Default value

0x0

cpuX.VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpuX.dcache-size

Set D-cache size in bytes

Type

int

Default value

0x8000

cpuX.dtcn_size

DTCM size in KB

Type

int

Default value

0x8

cpuX.icache-size

Set I-cache size in bytes

Type

int

Default value

0x8000

cpuX.itcm_size

ITCM size in KB

Type

int

Default value

0x8

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

cpuX.semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

cpuX.semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

cpuX.tcm-present

Disables the DTCM and ITCM

Type

bool

Default value

0x1

cpuX.vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether model has VFP support

Type

bool

Default value

0x1

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dic-spi_count

Number of shared peripheral interrupts implemented

Type

int

Default value

0x40

ecc_on

Enable Error Correcting Code

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

scheduler_mode

Control the interleaving of instructions in this processor. 0, default long quantum. 1, low latency mode, short quantum and signal checking. 2, lock-breaking mode, long quantum with additional context switches near load-exclusive instructions. WARNING: This parameter is intended for validation purposes and may result in unwanted behaviour if altered!

Type

int

Default value

0x0

3.5.64 ARM CortexR8x1CT

ARM CortexR8x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-445: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARM CortexR8x1CT

This model has the following Iris instances:

Table 3-446: ARM CortexR8x1CT Iris instances

InstanceName	ComponentName
ARM CortexR8x1CT	Cluster_ARM_Cortex-R8
ARM CortexR8x1CT.acp_mapper	PVBusMapper
ARM CortexR8x1CT.cpu0	ARM_Cortex-R8
ARM CortexR8x1CT.cpu0.l1dcache	PVCache
ARM CortexR8x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM CortexR8x1CT.cpu0.l1icache	PVCache
ARM CortexR8x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM CortexR8x1CT.ext_bus	PVBusLogger
ARM CortexR8x1CT.ext_bus.mapper	PVBusMapper
ARM CortexR8x1CT.l1_incoherent_interconnect	PVCache

InstanceName	ComponentName
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexR8x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-447: ARMCortexR8x1CT MTI instances

InstanceName	ComponentName
ARMCortexR8x1CT.acp_mapper	PVBusMapper
ARMCortexR8x1CT.cpu0	ARM_Cortex-R8
ARMCortexR8x1CT.cpu0.l1dcache	PVCache
ARMCortexR8x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR8x1CT.cpu0.l1icache	PVCache
ARMCortexR8x1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexR8x1CT.ext_bus	PVBusLogger
ARMCortexR8x1CT.ext_bus.mapper	PVBusMapper
ARMCortexR8x1CT.l1_incoherent_interconnect	PVCache
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[0]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[10]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[11]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[12]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[13]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[14]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[15]	PVBusSlave

InstanceName	ComponentName
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[16]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[17]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[2]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[3]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[4]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[5]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[6]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[7]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[8]	PVBusSlave
ARMCortexR8x1CT.l1_incoherent_interconnect.upstream[9]	PVBusSlave
ARMCortexR8x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexR8x1CT contains the following CADI targets:

- ARM_Cortex-R8
- Cluster_ARM_Cortex-R8
- PVCache

About ARMCortexR8x1CT

- The following components also exist:
 - ARMCortexR8x2CT.
 - ARMCortexR8x3CT.
 - ARMCortexR8x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

- `pvbus_s` is the slave port to access the TCM RAM of CPU *n*. Bits [3:0] of the user flags in the transaction are used to select the TCM:
 - 0 selects the ITCM of CPU 0.
 - 1 selects the DTCM of CPU 0.
 - 2 selects the ITCM of CPU 1.
 - 3 selects the DTCM of CPU 1.
 - 4 selects the ITCM of CPU 2.
 - 5 selects the DTCM of CPU 2.
 - 6 selects the ITCM of CPU 3.
 - 7 selects the DTCM of CPU 3.
 - Any other value is reserved.
- The `semihosting-cwd` parameter sets the CWD that is used for semihosting. The host operating system limits the maximum path length. The `semihosting-cwd` parameter does not provide any

security. Software running on the model can access files outside this directory using relative paths containing `..` or using absolute paths.

Differences between the model and the RTL

This component has the following differences from the corresponding revision of the RTL implementation:

- This component does not implement address filtering within the SCU. The enable bit for this feature is ignored.
- The GIC does not respect the CFGSDISABLE signal. This leads to some registers being accessible when they must not be.
- The SCU enable bit is ignored. The SCU is always enabled.
- The SCU ignores the invalidate all register.
- The Broadcast TLB or cache operations in this model do not cause other cores in the cluster that are asleep because of WFI to wake up.
- The RR bit in the SCTLR is ignored.
- The Power Control Register in the system control coprocessor is implemented but writing to it does not change the behavior of the model.
- The model cannot be configured with a 128-entry TLB.
- When modeling the SCU, coherency operations are represented by a memory write followed by a read to refill from memory, rather than using cache-to-cache transfers.
- TCMs are modeled internally and the model does not support external TCMs or the ports associated with them.
- This component implements L1 cache as architecturally defined, but does not implement L2 cache. If you require an L2 cache you can add a PL310 Level 2 Cache Controller component.
- The `vfp-enable_at_reset` option is model-specific and has no hardware equivalent. Arm recommends that it is only used in test systems and tied off to `false` in production systems.
- ECC and parity schemes are hardware-specific so are not supported.

Ports for ARMCortexR8x1CT

Table 3-448: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>cfgend[1]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgnmfi[1]</code>	Signal	Slave	This signal disables FIQ mask in CPSR.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>clusterid</code>	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
<code>event</code>	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
<code>fiq[1]</code>	Signal	Slave	Legacy FIQ request input line.
<code>fiqout[1]</code>	Signal	Master	Output of individual processor nFIQ from the interrupt controller.
<code>fpfilterend0</code>	Value	Slave	This port sets end of region mapped to <code>pvbus_mfp0</code> .

Name	Protocol	Type	Description
fpfilterend1	Value	Slave	This port sets end of region mapped to pvbus_mfp1.
fpfilterend2	Value	Slave	This port sets end of region mapped to pvbus_mfp2.
fpfilterend3	Value	Slave	This port sets end of region mapped to pvbus_mfp3.
fpfilterstart0	Value	Slave	This port sets start of region mapped to pvbus_mfp0.
fpfilterstart1	Value	Slave	This port sets start of region mapped to pvbus_mfp1.
fpfilterstart2	Value	Slave	This port sets start of region mapped to pvbus_mfp2.
fpfilterstart3	Value	Slave	This port sets start of region mapped to pvbus_mfp3.
fpuflags[1]	ValueState	Master	Floating-Point Unit output flags.
halt[1]	Signal	Slave	Raising this signal will put the core into halt mode. Equivalent to the hardware nCPUHALT[N:0] signal.
initram[1]	Signal	Slave	This signal enables the processor to boot from the instruction TCM.
ints[480]	Signal	Slave	Interrupt distributor interrupt lines.
irq[1]	Signal	Slave	Legacy IRQ request input line.
irqout[1]	Signal	Master	Output of individual processor nIRQ from the interrupt controller.
mfilteren	Signal	Slave	This signal enables filtering of address ranges between master bus ports.
mfilterend	Value	Slave	This port sets end of region mapped to pvbus_m1.
mfilterstart	Value	Slave	This port sets start of region mapped to pvbus_m1.
periphbase	Value	Slave	This port sets the base address of private peripheral region.
periphclk_in	ClockSignal	Slave	The timer and the watchdog take need a clk that is scaled down at least by factor of two.
periphreset	Signal	Slave	This signal resets timer and interrupt controller.
pfilterend	Value	Slave	This port sets end of region mapped to pvbus_mp.
pfilterstart	Value	Slave	This port sets start of region mapped to pvbus_mp.
pmuirq[2]	Signal	Master	Interrupt signal from performance monitoring unit.
pvbus_m0	PVBus	Master	AXI master port 0.
pvbus_m1	PVBus	Master	AXI master port 1.
pvbus_mfp0	PVBus	Master	Fast peripheral port for core 0.
pvbus_mfp1	PVBus	Master	Fast peripheral port for core 1.
pvbus_mfp2	PVBus	Master	Fast peripheral port for core 2.
pvbus_mfp3	PVBus	Master	Fast peripheral port for core 3.
pvbus_mp	PVBus	Master	Shared peripheral port.
pvbus_s	PVBus	Slave	tcm slave port.
reset[1]	Signal	Slave	Raising this signal will put the core into reset mode.
scureset	Signal	Slave	This signal resets SCU.
smpnamp[1]	Signal	Master	This signals AMP or SMP mode for each Cortex-R8 processor.
standbywfe[1]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[1]	Signal	Master	This signal indicates if a core is in WFI state.
teinit[1]	Signal	Slave	This signal provides default exception handling state.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.

Name	Protocol	Type	Description
vinithi[1]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
wdreset[1]	Signal	Slave	This signal resets individual watchdog.
wdresetreq[1]	Signal	Master	CPU watchdog reset requests.

Parameters for ARM Cortex R8x1CT

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

FPFILTEREND0

Specifies the end address for the fast peripheral port address filtering

Type

int

Default value

0x0

FPFILTEREND1

Specifies the end address for the fast peripheral port address filtering

Type

int

Default value

0x0

FPFILTEREND2

Specifies the end address for the fast peripheral port address filtering

Type

int

Default value

0x0

FPFILTEREND3

Specifies the end address for the fast peripheral port address filtering

Type

int

Default value

0x0

FPFILTERSTART0

Specifies the start address for the fast peripheral port address filtering

Type

int

Default value

0xffff00000

FPFILTERSTART1

Specifies the start address for the fast peripheral port address filtering

Type

int

Default value

0xffff00000

FPFILTERSTART2

Specifies the start address for the fast peripheral port address filtering

Type

int

Default value

0xffff00000

FPFILTERSTART3

Specifies the start address for the fast peripheral port address filtering

Type

int

Default value

0xffff00000

LOCK_STEP

Affects dual-processor configurations only, and ignored by single-processor configurations. 0 - Disable. Set for two independent processors. 1 - Lock Step. Appears to the system as two processors but is internally modeled as a single processor. 3 - Split Lock. Appears to the system as two processors but can be statically configured from reset either as two independent processors or two locked processors. For the model, these are equivalent to Disable and Lock Step, respectively, except for the value of build options registers. The model does not support dynamically splitting and locking the processor.

Type

int

Default value

0x0

MFILTEREN

Enables filtering of address ranges

Type

bool

Default value

0x0

MFILTEREND

Specifies the end address for address filtering

Type

int

Default value

0x0

MFILTERSTART

Specifies the start address for address filtering

Type

int

Default value

0x0

NUM_MPU_REGION

Sets the number of MPU regions

Type

int

Default value

0xc

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0xae000000

PFILTEREND

Specifies the end address for peripheral port address filtering

Type

int

Default value

0x0

PFILTERSTART

Specifies the start address for peripheral port address filtering

Type

int

Default value

0xffff00000

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Initialize to BE8 endianness

Type

bool

Default value

0x0

cpuX.CFGNMFI

Enable nonmaskable FIQ interrupts on startup

Type

bool

Default value

0x0

cpuX.DP_FLOAT

Sets whether double-precision instructions are available

Type

bool

Default value

0x1

cpuX.INITRAM

Enable the processor to boot from the instruction TCM

Type

bool

Default value

0x0

cpuX.POWERCTLI

Default power control state for processor

Type

int

Default value

0x0

cpuX.SMPnAMP

Set whether the processor is part of a coherent domain

Type

bool

Default value

0x0

cpuX.TEINIT

T32 exception enable. The default has exceptions including reset handled in A32 state

Type

bool

Default value

0x0

cpuX.VINITHI

Initialize with high vectors enabled

Type

bool

Default value

0x0

cpuX.dcache-size

Set D-cache size in bytes

Type

int

Default value

0x8000

cpuX.dtcn_size

DTCM size in KB

Type

int

Default value

0x8

cpuX.icache-size

Set I-cache size in bytes

Type

int

Default value

0x8000

cpuX.itcm_size

ITCM size in KB

Type

int

Default value

0x8

cpuX.min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-ARM_HLT

ARM HLT number for semihosting

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

ARM SVC number for semihosting

Type

int

Default value

0x123456

cpuX.semihosting-Thumb_HLT

Thumb HLT number for semihosting

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

Thumb SVC number for semihosting

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0xf000000

cpuX.semihosting-hlt-enable

Enable semihosting HLT traps. Applications that use HLT semihosting must set this parameter to true and the semihosting-enable parameter to true

Type

bool

Default value

0x0

cpuX.semihosting-prefix

Prefix semihosting output with target instance name

Type

bool

Default value

0x0

cpuX.semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0xf000000

cpuX.tcm-present

Disables the DTCM and ITCM

Type

bool

Default value

0x1

cpuX.vfp-enable_at_reset

Enable coprocessor access and VFP at reset

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether model has VFP support

Type

bool

Default value

0x1

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dic-spi_count

Number of shared peripheral interrupts implemented

Type

int

Default value

0x40

ecc_on

Enable Error Correcting Code

Type

bool

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

scheduler_mode

Control the interleaving of instructions in this processor. 0, default long quantum. 1, low latency mode, short quantum and signal checking. 2, lock-breaking mode, long quantum with additional context switches near load-exclusive instructions. WARNING: This parameter is intended for validation purposes and may result in unwanted behaviour if altered!

Type

int

Default value

0x0

3.5.65 ARM CortexR52PlusCT

ARM CortexR52PlusCT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-449: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARM CortexR52PlusCT

This model has the following Iris instances:

Table 3-450: ARM CortexR52PlusCT Iris instances

InstanceName	ComponentName
ARM CortexR52PlusCT	Cluster_ARM_Cortex-R52Plus
ARM CortexR52PlusCT.AMU	PVBusLogger
ARM CortexR52PlusCT.AMU.mapper	PVBusMapper
ARM CortexR52PlusCT.DAP	PVBusLogger
ARM CortexR52PlusCT.DAP.mapper	PVBusMapper
ARM CortexR52PlusCT.MMAP	PVBusLogger
ARM CortexR52PlusCT.MMAP.mapper	PVBusMapper
ARM CortexR52PlusCT.acp_mapper	PVBusMapper
ARM CortexR52PlusCT.cpu0	ARM_Cortex-R52Plus
ARM CortexR52PlusCT.cpu0.UTLB	TLB

InstanceName	ComponentName
ARMCortexR52PlusCT.cpu0.dtlb	TlbCadi
ARMCortexR52PlusCT.cpu0.gicv3_cpu_if	GICv3CPUInterface
ARMCortexR52PlusCT.cpu0.l1dcache	PVCache
ARMCortexR52PlusCT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR52PlusCT.cpu0.l1icache	PVCache
ARMCortexR52PlusCT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexR52PlusCT.ext_bus	PVBusLogger
ARMCortexR52PlusCT.ext_bus.mapper	PVBusMapper
ARMCortexR52PlusCT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexR52PlusCT.gic_iri	GIC_IRI
ARMCortexR52PlusCT.gic_iri.rd_0	GICv3RedistributorInternal
ARMCortexR52PlusCT.gic_iri.rd_0_0	GICv3RedistributorInternal
ARMCortexR52PlusCT.gic_iri.rd_0_0_0	GICv3RedistributorInternal
ARMCortexR52PlusCT.gic_iri.rd_0_0_0_0	GICv3Redistributor
ARMCortexR52PlusCT.gic_iri.rd_0_0_0_1	GICv3Redistributor
ARMCortexR52PlusCT.gic_iri.rd_t1	GICv3Distributor
ARMCortexR52PlusCT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-451: ARMCortexR52PlusCT MTI instances

InstanceName	ComponentName
ARMCortexR52PlusCT	ARMv8Cluster
ARMCortexR52PlusCT.AMU	PVBusLogger
ARMCortexR52PlusCT.AMU.mapper	PVBusMapper
ARMCortexR52PlusCT.DAP	PVBusLogger
ARMCortexR52PlusCT.DAP.mapper	PVBusMapper
ARMCortexR52PlusCT.MMAP	PVBusLogger
ARMCortexR52PlusCT.MMAP.mapper	PVBusMapper
ARMCortexR52PlusCT.acp_mapper	PVBusMapper
ARMCortexR52PlusCT.cpu0	ARM_Cortex-R52Plus
ARMCortexR52PlusCT.cpu0.UTLB	TLB
ARMCortexR52PlusCT.cpu0.gicv3_cpu_if	GICv3CPUInterface
ARMCortexR52PlusCT.cpu0.l1dcache	PVCache
ARMCortexR52PlusCT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR52PlusCT.cpu0.l1icache	PVCache
ARMCortexR52PlusCT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexR52PlusCT.ext_bus	PVBusLogger
ARMCortexR52PlusCT.ext_bus.mapper	PVBusMapper
ARMCortexR52PlusCT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

InstanceName	ComponentName
ARMCortexR52PlusCT.gic_iri	GICv3IRI
ARMCortexR52PlusCT.gic_iri.rd_0	GICv3RedistributorInternal
ARMCortexR52PlusCT.gic_iri.rd_0_0	GICv3RedistributorInternal
ARMCortexR52PlusCT.gic_iri.rd_0_0_0	GICv3RedistributorInternal
ARMCortexR52PlusCT.gic_iri.rd_0_0_0_0	GICv3Redistributor
ARMCortexR52PlusCT.gic_iri.rd_0_0_0_1	GICv3Redistributor
ARMCortexR52PlusCT.gic_iri.rd_tl	GICv3Distributor
ARMCortexR52PlusCT.l2_flusher	AsyncCacheFlushUnit

ARMCortexR52PlusCT contains the following CADI targets:

- ARM_Cortex-R52Plus
- Cluster_ARM_Cortex-R52Plus
- PVCache
- TlbCadi
- gic_iri

About ARMCortexR52PlusCT

The model does not implement the following:

- Redundant cores for Dual Core Lock Step operation.
- Low Power Interface to wake the target core on receiving a `wake_request` from the GIC Distributor.

The following models also exist:

- ARMCortex-R52Plusx1CT
- ARMCortex-R52Plusx2CT
- ARMCortex-R52Plusx3CT
- ARMCortex-R52Plusx4CT

Debug accesses

For debug accesses to succeed, they must have permissions that are compliant with the permission value in the `IMP_SLAVEPCTLR` register.



The reset value of `IMP_SLAVEPCTLR` is `0x1` which means privileged access only.

Ports for ARMCortexR52PlusCT

Table 3-452: Ports

Name	Protocol	Type	Description
cfgdbgromaddr	Value_64	Slave	Debug ROM base address.
cfgdbgromaddrv	Signal	Slave	Debug ROM base address valid.
cfgendianess[4]	Signal	Slave	This signal if for EE bit initialisation.
cfgperiphbase	Value_64	Slave	This port sets the base address of private peripheral region.
cfgthumbexceptions[4]	Signal	Slave	This signal provides default exception handling state.
cfgvectable[4]	Value_64	Slave	Reset vector base address.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clusterid	Value	Slave	Configure Aff2 and Aff1 fields of MPIDR. Aff1 = value[7:0], Aff2 = value[15:8]
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
commr[4]	Signal	Master	Receive portion of Data Transfer Register full.
commtx[4]	Signal	Master	Transmit portion of Data Transfer Register empty.
cpuhalt[4]	Signal	Slave	Raising this signal will put the core into halt mode.
cpuporeset[4]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
dbgack[4]	Signal	Master	External debug interface.
dbgen[4]	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	There is no support for PChannel in CortexR52Plus. These signals relate to core power down. Equivalent to COREPACTIVEEx
dbgpwrupreq[4]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
edbgrq[4]	Signal	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
ext_slave_s	PVBus	Slave	External Slave port. Equivalent to AXIS port
extppi_in_0[9]	Signal	Slave	Core 0 external ppi signals.
extppi_in_1[9]	Signal	Slave	Core 1 external ppi signals.
extppi_in_2[9]	Signal	Slave	Core 2 external ppi signals.
extppi_in_3[9]	Signal	Slave	Core 3 external ppi signals.
flash_m[4]	PVBus	Master	Flash Port.
gdu_external_m	GICv3Comms	Master	GDU external messaging port.
hiden[4]	Signal	Slave	External debug interface.

Name	Protocol	Type	Description
hniden[4]	Signal	Slave	External debug interface.
llpp_m[4]	PVBus	Master	LLPP (Low-Latency Peripheral Port).
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[4]	Signal	Slave	External debug interface.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_core_m[4]	PVBus	Master	The core will generate bus requests on this port. Equivalent to AXIM port
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
sei[4]	Signal	Slave	Per core virtual System Error physical pins.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
standbywfe[4]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[4]	Signal	Master	This signal indicates if a core is in WFI state.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
topreset	Signal	Slave	This signal resets timer and interrupt controller.
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.
warmrstreq[4]	Signal	Master	Warm reset request from core.

Parameters for ARMCortexR52PlusCT

CLUSTER_ID

CLUSTER_ID[15:8] equivalent to CFGMPIDRAFF2, CLUSTER_ID[7:0] equivalent to CFGMPIDRAFF1

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Equivalent to CFGDBGROMADDR

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x0

NUM_CORES

Number of cores in cluster

Type

int

Default value

0x1

PERIPHBASE

Equivalent to CFGPERIPHBASE

Type

int

Default value

0x13080000

cluster_utid

Equivalent to CFGCLUSTERUTID

Type

int

Default value

0x0

cpu_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Equivalent to CFGTHUMBEXCEPTIONS

Type

bool

Default value

0x0

cpuX.RVBARADDR

Equivalent to CFGVECTABLE

Type

int

Default value

0x0

cpuX.ase-present

Set whether the model has been built with NEON support

Type

bool

Default value

0x1

cpuX.dcache-size

L1 D-Cache size in bytes

Type

int

Default value

0x8000

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.flash.enable

Equivalent to CFGFLASHEN

Type

bool

Default value

0x0

cpuX.icache-size

L1 I-Cache size in bytes

Type

int

Default value

0x8000

cpuX.llpp.base

Equivalent to CFGLLPPBASEADDR

Type

int

Default value

0x0

cpuX.llpp.size

Equivalent to CFGLLPPSIZE

Type

int

Default value

0x8000000

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.tcm.a.enable

Equivalent to CFGTCMBOOT

Type

bool

Default value

0x0

cpuX.tcm.a.size

Sets the size of the ATCM(in bytes)

Type

int

Default value

0x4000

cpuX.tcm.a.wait

TCM Register A accesses wait states: 0-1 states

Type

int

Default value

0x0

cpuX.tcm.b.size

Sets the size of the BTCM(in bytes)

Type

int

Default value

0x4000

cpuX.tcm.b.wait

TCM Register B accesses wait states: 0-1 states

Type

int

Default value

0x0

cpuX.tcm.c.size

Sets the size of the CTCM(in bytes)

Type

int

Default value

0x2000

cpuX.tcm.c.wait

TCM Register C accesses wait states: 0-1 states

Type

int

Default value

0x0

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-dp-present

Whether double-precision floating point feature is implemented (FEAT_F64MM).

Type

bool

Default value

0x1

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_lock_step

(equivalent to CFGSLSPLIT)

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

flash_protection_enable_at_reset

Equivalent to CFGFLASHPROTEN

Type

bool

Default value

0x0

has_export_m_port

The interrupt distributor has an optional interrupt export port for routing interrupts to an external device

Type

bool

Default value

0x1

has_flash

Equivalent to CFGFLASHIMP

Type

bool

Default value

0x0

has_flash_protection

Equivalent to CFGFLASHPROTIMP

Type

bool

Default value

0x1

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

memory.ext_slave_base

Equivalent to CFGAXISTCMBASEADDR

Type

int

Default value

0x0

memory.flash_base

Equivalent to CFGFLASHBASEADDR

Type

int

Default value

0x0

num_protection_regions_s1

Number of v8-R protection regions

Type

int

Default value

0x18

num_protection_regions_s2

Number of v8-R hyp protection regions

Type

int

Default value

0x18

num_spi

Number of interrupts (SPI) into the internal GIC controller

Type

int

Default value

0x3c0

ram_protection_enable_at_reset

Equivalent to CFGRAMPROTEN

Type

bool

Default value

0x0

reported_fp_revision

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored. Updates the FPSID.revision value

Type

int

Default value

0xffffffffffffffff

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

3.5.66 ARM CortexR52x1CT

ARM CortexR52x1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-453: IP revisions support

Revision	Quality level
r1p4	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARM CortexR52x1CT

This model has the following Iris instances:

Table 3-454: ARM CortexR52x1CT Iris instances

InstanceName	ComponentName
ARM CortexR52x1CT	Cluster_ARM_Cortex-R52
ARM CortexR52x1CT.AMU	PVBusLogger
ARM CortexR52x1CT.AMU.mapper	PVBusMapper
ARM CortexR52x1CT.DAP	PVBusLogger
ARM CortexR52x1CT.DAP.mapper	PVBusMapper
ARM CortexR52x1CT.MMAP	PVBusLogger
ARM CortexR52x1CT.MMAP.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexR52x1CT.acp_mapper	PVBusMapper
ARMCortexR52x1CT.cpu0	ARM_CortexR52
ARMCortexR52x1CT.cpu0.UTLB	TLB
ARMCortexR52x1CT.cpu0.dtlb	TlbCadi
ARMCortexR52x1CT.cpu0.gicv3_cpu_if	GICv3CPUInterface
ARMCortexR52x1CT.cpu0.l1dcache	PVCache
ARMCortexR52x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR52x1CT.cpu0.l1licache	PVCache
ARMCortexR52x1CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexR52x1CT.ext_bus	PVBusLogger
ARMCortexR52x1CT.ext_bus.mapper	PVBusMapper
ARMCortexR52x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexR52x1CT.gic_iri	GIC_IRI
ARMCortexR52x1CT.gic_iri.rd_0	GICv3RedistributorInternal
ARMCortexR52x1CT.gic_iri.rd_0_0	GICv3RedistributorInternal
ARMCortexR52x1CT.gic_iri.rd_0_0_0	GICv3RedistributorInternal
ARMCortexR52x1CT.gic_iri.rd_0_0_0_0	GICv3Redistributor
ARMCortexR52x1CT.gic_iri.rd_0_0_0_1	GICv3Redistributor
ARMCortexR52x1CT.gic_iri.rd_t1	GICv3Distributor
ARMCortexR52x1CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-455: ARMCortexR52x1CT MTI instances

InstanceName	ComponentName
ARMCortexR52x1CT	ARMv8Cluster
ARMCortexR52x1CT.AMU	PVBusLogger
ARMCortexR52x1CT.AMU.mapper	PVBusMapper
ARMCortexR52x1CT.DAP	PVBusLogger
ARMCortexR52x1CT.DAP.mapper	PVBusMapper
ARMCortexR52x1CT.MMAP	PVBusLogger
ARMCortexR52x1CT.MMAP.mapper	PVBusMapper
ARMCortexR52x1CT.acp_mapper	PVBusMapper
ARMCortexR52x1CT.cpu0	ARM_CortexR52
ARMCortexR52x1CT.cpu0.UTLB	TLB
ARMCortexR52x1CT.cpu0.gicv3_cpu_if	GICv3CPUInterface
ARMCortexR52x1CT.cpu0.l1dcache	PVCache
ARMCortexR52x1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR52x1CT.cpu0.l1licache	PVCache
ARMCortexR52x1CT.cpu0.l1licache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexR52x1CT.ext_bus	PVBusLogger
ARMCortexR52x1CT.ext_bus.mapper	PVBusMapper
ARMCortexR52x1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexR52x1CT.gic_iri	GICv3IRI
ARMCortexR52x1CT.gic_iri.rd_0	GICv3RedistributorInternal
ARMCortexR52x1CT.gic_iri.rd_0_0	GICv3RedistributorInternal
ARMCortexR52x1CT.gic_iri.rd_0_0_0	GICv3RedistributorInternal
ARMCortexR52x1CT.gic_iri.rd_0_0_0_0	GICv3Redistributor
ARMCortexR52x1CT.gic_iri.rd_0_0_0_1	GICv3Redistributor
ARMCortexR52x1CT.gic_iri.rd_tl	GICv3Distributor
ARMCortexR52x1CT.l2_flusher	AsyncCacheFlushUnit

ARMCortexR52x1CT contains the following CADI targets:

- ARM_CortexR52
- Cluster_ARM_Cortex-R52
- PVCache
- TlbCadi
- gic_iri

About ARMCortexR52x1CT

- The following components also exist:
 - ARMCortexR52x2CT.
 - ARMCortexR52x3CT.
 - ARMCortexR52x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

- When you use the `semihosting-cmd_line` parameter to set the command line that is available to semihosting SVC calls, the value of `argv[0]` is set to the first command-line argument, not to the name of an image.
- The Cortex-R52 processor does not implement TrustZone® technology, therefore the model does not support `S_*` or `NS_*` registers or exceptions.
- If flash memory is not enabled, to disable all routing to the flash port, set the `has_flash` parameter to false.

Differences between the model and the RTL

- The model does not implement redundant cores for Dual-Core Lock-Step operations.
- The model does not implement the Low Power Interface to wake up the target core on receiving a `wake_request` signal from the GIC distributor.

- TCMs are modeled internally and the model does not support external TCMs or the ports associated with them.
- The model does not support running Software Test Libraries (STLs).
- The `vfp-enable_at_reset` parameter is a model-specific behavior with no hardware equivalent.
- ECC and parity schemes are hardware-specific so are not supported.

Debug accesses

For debug accesses to succeed, they must have permissions that are compliant with the permission value in the IMP_SLAVEPCTLR register.



The reset value of IMP_SLAVEPCTLR is 0x1 which means privileged access only.

Ports for ARMCortexR52x1CT

Table 3-456: Ports

Name	Protocol	Type	Description
<code>cfgdbgromaddr</code>	Value_64	Slave	Debug ROM base address.
<code>cfgdbgromaddrv</code>	Signal	Slave	Debug ROM base address valid.
<code>cfgendianess[1]</code>	Signal	Slave	This signal if for EE bit initialisation.
<code>cfgperiphbase</code>	Value_64	Slave	This port sets the base address of private peripheral region.
<code>cfgthumbexceptions[1]</code>	Signal	Slave	This signal provides default exception handling state.
<code>cfgvectable[1]</code>	Value_64	Slave	Reset vector base address.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.
<code>clrexmonack</code>	Signal	Master	Acknowledge handshake signal for the <code>clrexmonreq</code> signal
<code>clrexmonreq</code>	Signal	Slave	Signals the clearing of an external global exclusive monitor
<code>clusterid</code>	Value	Slave	Configure Aff2 and Aff1 fields of MPIDR. Aff1 = value[7:0], Aff2 = value[15:8]
<code>cntvalueb</code>	CounterInterface	Slave	Interface to SoC level counter module.
<code>commr[1]</code>	Signal	Master	Receive portion of Data Transfer Register full.
<code>commtx[1]</code>	Signal	Master	Transmit portion of Data Transfer Register empty.
<code>COREPACTIVEx1[1]</code>	Signal	Master	These signals relate to core power down. Equivalent to COREPACTIVEx[1]
<code>cpuhalt[1]</code>	Signal	Slave	Raising this signal will put the core into halt mode.

Name	Protocol	Type	Description
cpuporeset[1]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cti[1]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
dbgack[1]	Signal	Master	External debug interface.
dbgen[1]	Signal	Slave	External debug interface.
dbgnopwrdown[1]	Signal	Master	There is no support for PChannel in CortexR52. These signals relate to core power down. Equivalent to COREPACTIVEx
dbgprupreq[1]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
edbgrq[1]	Signal	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
ext_slave_s	PVBus	Slave	External Slave port. Equivalent to AXIS port
extppi_in_0[9]	Signal	Slave	Core 0 external ppi signals.
flash_m[1]	PVBus	Master	Flash Port.
gdu_external_m	GICv3Comms	Master	GDU external messaging port.
hiden[1]	Signal	Slave	External debug interface.
hniden[1]	Signal	Slave	External debug interface.
llpp_m[1]	PVBus	Master	LLPP (Low-Latency Peripheral Port).
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden[1]	Signal	Slave	External debug interface.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_core_m[1]	PVBus	Master	The core will generate bus requests on this port. Equivalent to AXIM port
reset[1]	Signal	Slave	Raising this signal will put the core into reset mode.
sei[1]	Signal	Slave	Per core virtual System Error physical pins.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
standbywfe[1]	Signal	Master	This signal indicates if a core is in WFE state.
standbywfi[1]	Signal	Master	This signal indicates if a core is in WFI state.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
topreset	Signal	Slave	This signal resets timer and interrupt controller.
vsei[1]	Signal	Slave	Per core virtual System Error physical pins.
warmrstreq[1]	Signal	Master	Warm reset request from core.

Parameters for ARM Cortex R52x1CT

CLUSTER_ID

CLUSTER_ID[15:8] equivalent to CFGMPIDRAFF2, CLUSTER_ID[7:0] equivalent to CFGMPIDRAFF1

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Equivalent to CFGDBGROMADDR

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x0

PERIPHBASE

Equivalent to CFGPERIPHBASE

Type

int

Default value

0x13080000

cluster_utid

Equivalent to CFGCLUSTERUTID

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Equivalent to CFGTHUMBEXCEPTIONS

Type

bool

Default value

0x0

cpuX.rvbaraddr

Equivalent to CFGVECTABLE

Type

int

Default value

0x0

cpuX.ase-present

Set whether the model has been built with NEON support

Type

bool

Default value

0x1

cpuX.dcache-size

L1 D-Cache size in bytes

Type

int

Default value

0x8000

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.flash.enable

Equivalent to CFGFLASHEN

Type

bool

Default value

0x0

cpuX.icache-size

L1 I-Cache size in bytes

Type

int

Default value

0x8000

cpuX.llpp.base

Equivalent to CFGLLPPBASEADDR

Type

int

Default value

0x0

cpuX.llpp.size

Equivalent to CFGLLPPSIZE

Type

int

Default value

0x1000

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.tcm.a.enable

Equivalent to CFGTCMBOOT

Type

bool

Default value

0x0

cpuX.tcm.a.size

Sets the size of the ATCM(in bytes)

Type

int

Default value

0x4000

cpuX.tcm.a.wait

TCM Register A accesses wait states: 0-1 states

Type

int

Default value

0x0

cpuX.tcm.b.size

Sets the size of the BTCM(in bytes)

Type

int

Default value

0x4000

cpuX.tcm.b.wait

TCM Register B accesses wait states: 0-1 states

Type

int

Default value

0x0

cpuX.tcm.c.size

Sets the size of the CTCM(in bytes)

Type

int

Default value

0x2000

cpuX.tcm.c.wait

TCM Register C accesses wait states: 0-1 states

Type

int

Default value

0x0

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-dp-present

Whether double-precision floating point feature is implemented (FEAT_F64MM).

Type

bool

Default value

0x1

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_lock_step

(equivalent to CFGSLSPPLIT)

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

flash_protection_enable_at_reset

Equivalent to CFGFLASHPROTEN

Type

bool

Default value

0x0

has_export_m_port

The interrupt distributor has an optional interrupt export port for routing interrupts to an external device

Type

bool

Default value

0x1

has_flash

Equivalent to CFGFLASHIMP

Type

bool

Default value

0x0

has_flash_protection

Equivalent to CFGFLASHPROTIMP

Type

bool

Default value

0x1

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

memory.ext_slave_base

Equivalent to CFGAXISTCMBASEADDR

Type

int

Default value

0x0

memory.flash_base

Equivalent to CFGFLASHBASEADDR

Type

int

Default value

0x0

num_protection_regions_s1

Number of v8-R protection regions

Type

int

Default value

0x18

num_protection_regions_s2

Number of v8-R hyp protection regions

Type

int

Default value

0x18

num_spi

Number of interrupts (SPI) into the internal GIC controller

Type

int

Default value

0x3c0

ram_protection_enable_at_reset

Equivalent to CFGRAMPROTEN

Type

bool

Default value

0x0

reported_fp_revision

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored. Updates the FPSID.revision value

Type

int

Default value

0xffffffffffffffff

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

3.5.67 ARM Cortex R82AECT

ARM Cortex R82AECT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-457: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- CHI

Parameters removed:

- bus_type

Iris and MTI instances for ARM CortexR82AECT

This model has the following Iris instances:

Table 3-458: ARM CortexR82AECT Iris instances

InstanceName	ComponentName
ARM CortexR82AECT	Cluster_ARM_Cortex-R82AE
ARM CortexR82AECT.AMU	PVBusLogger
ARM CortexR82AECT.AMU.mapper	PVBusMapper
ARM CortexR82AECT.DAP	PVBusLogger
ARM CortexR82AECT.DAP.mapper	PVBusMapper
ARM CortexR82AECT.DSU	DSU
ARM CortexR82AECT.DSU.PPU_cluster	PPUv1
ARM CortexR82AECT.DSU.PPU_cluster.busslave	PVBusSlave
ARM CortexR82AECT.DSU.PPU_core0	PPUv1
ARM CortexR82AECT.DSU.PPU_core0.busslave	PVBusSlave
ARM CortexR82AECT.DSU.utility_slave[0]	PVBusSlave
ARM CortexR82AECT.MMAP	PVBusLogger
ARM CortexR82AECT.MMAP.mapper	PVBusMapper
ARM CortexR82AECT.cpu0	ARM_Cortex-R82AE
ARM CortexR82AECT.cpu0.UTLB	TLB
ARM CortexR82AECT.cpu0.dtlb	TlbCadi
ARM CortexR82AECT.cpu0.gicv3_cpu_if	GICv3CPUInterface
ARM CortexR82AECT.cpu0.l1dcache	PVCache
ARM CortexR82AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM CortexR82AECT.cpu0.l1licache	PVCache
ARM CortexR82AECT.cpu0.l1licache.upstream[0]	PVBusSlave
ARM CortexR82AECT.ext_bus	PVBusLogger
ARM CortexR82AECT.ext_bus.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexR82AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexR82AECT.l2_cache	PVCache
ARMCortexR82AECT.l2_cache.upstream[0]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[10]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[11]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[12]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[13]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[14]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[15]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[16]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[1]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[2]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[3]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[4]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[5]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[6]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[7]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[8]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[9]	PVBusSlave
ARMCortexR82AECT.llram_atomic_filter	LLRAMAtomicOpFilter
ARMCortexR82AECT.llram_atomic_filter.atomic_bus_slave	PVBusSlave
ARMCortexR82AECT.llram_atomic_filter.filter	PVBusMapper
ARMCortexR82AECT.llram_atomic_filter.llram_exclusive_monitor	PVBusExclusiveMonitor
ARMCortexR82AECT.llram_atomic_filter.llram_exclusive_monitor.bus_mapper	PVBusMapper
ARMCortexR82AECT.llram_atomic_filter.pvbusmaster	PVBusMaster
ARMCortexR82AECT.llram_coherent_interconnect	PVCache
ARMCortexR82AECT.llram_coherent_interconnect.upstream[0]	PVBusSlave
ARMCortexR82AECT.llram_coherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR82AECT.llram_coherent_interconnect.upstream[2]	PVBusSlave

This model has the following MTI trace components:

Table 3-459: ARMCortexR82AECT MTI instances

InstanceName	ComponentName
ARMCortexR82AECT	ARMv8Cluster
ARMCortexR82AECT.AMU	PVBusLogger
ARMCortexR82AECT.AMU.mapper	PVBusMapper
ARMCortexR82AECT.DAP	PVBusLogger
ARMCortexR82AECT.DAP.mapper	PVBusMapper
ARMCortexR82AECT.DSU	DSU

InstanceName	ComponentName
ARMCortexR82AECT.DSU.PPU_cluster	PPUv1
ARMCortexR82AECT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexR82AECT.DSU.PPU_core0	PPUv1
ARMCortexR82AECT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexR82AECT.DSU.utility_slave[0]	PVBusSlave
ARMCortexR82AECT.MMAP	PVBusLogger
ARMCortexR82AECT.MMAP.mapper	PVBusMapper
ARMCortexR82AECT.cpu0	ARM_Cortex-R82AE
ARMCortexR82AECT.cpu0.UTLB	TLB
ARMCortexR82AECT.cpu0.gicv3_cpu_if	GICv3CPUInterface
ARMCortexR82AECT.cpu0.l1dcache	PVCache
ARMCortexR82AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR82AECT.cpu0.l1licache	PVCache
ARMCortexR82AECT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexR82AECT.ext_bus	PVBusLogger
ARMCortexR82AECT.ext_bus.mapper	PVBusMapper
ARMCortexR82AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexR82AECT.l2_cache	PVCache
ARMCortexR82AECT.l2_cache.upstream[0]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[10]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[11]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[12]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[13]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[14]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[15]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[16]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[1]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[2]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[3]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[4]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[5]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[6]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[7]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[8]	PVBusSlave
ARMCortexR82AECT.l2_cache.upstream[9]	PVBusSlave
ARMCortexR82AECT.llram_atomic_filter	LLRAMAtomicOpFilter
ARMCortexR82AECT.llram_atomic_filter.atomic_bus_slave	PVBusSlave
ARMCortexR82AECT.llram_atomic_filter.filter	PVBusMapper
ARMCortexR82AECT.llram_atomic_filter.llram_exclusive_monitor	PVBusExclusiveMonitor

InstanceName	ComponentName
ARMCortexR82AECT.llram_atomic_filter.llram_exclusive_monitor.bus_mapper	PVBusMapper
ARMCortexR82AECT.llram_atomic_filter.pvbusmaster	PVBusMaster
ARMCortexR82AECT.llram_coherent_interconnect	PVCache
ARMCortexR82AECT.llram_coherent_interconnect.upstream[0]	PVBusSlave
ARMCortexR82AECT.llram_coherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR82AECT.llram_coherent_interconnect.upstream[2]	PVBusSlave

ARMCortexR82AECT contains the following CADI targets:

- ARM_Cortex-R82AE
- Cluster_ARM_Cortex-R82AE
- PVCache
- TlbCadi

AE-specific features implemented

The only AE-specific features implemented in the model are Split-Lock and Lock-Step, with the following limitations:

- The implementation of Split-Lock and Lock-Step in the model is simply a halving of the number of cores available.
- Lock-step is implemented by a parameter that can be true or false for the entire execution only, rather than by a signal that can be enabled during execution.

As the model does not implement true Split-Lock functionality, the ports and parameters are contiguous. For example, for a 4-core CPU with Split-Lock enabled, `cpu0` and `cpu1` identify the available cores and associated ports, not `cpu0` and `cpu2`.

- Hybrid mode is not modeled in the DSU.

Ports for ARMCortexR82AECT

Table 3-460: Ports

Name	Protocol	Type	Description
<code>acel_s</code>	PVBus	Slave	External Slave port. Equivalent to AXIS port.
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	Enable broadcasting of cache maintenance operations to downstream caches.
<code>broadcastouter</code>	Signal	Slave	Enable broadcasting of Outer Shareable transactions.
<code>cache_validation_control</code>	Value	Slave	This signal provides default exception handling state.
<code>cfgendianess[8]</code>	Signal	Slave	This signal is for EE bit initialisation
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the core executes instructions.

Name	Protocol	Type	Description
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPSIRQ[8]	Signal	Master	Timer signals to SOC
CNTHVSIRQ[8]	Signal	Master	Timer signals to SOC
CNTPSIRQ[8]	Signal	Master	Timer signals to SOC
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module
CNTVIRQ[8]	Signal	Master	Timer signals to SOC
commirq[8]	Signal	Master	Interrupt signal from debug communication channel.
core_pcs_m_pchannel[8]	PChannel	Master	Core PCSM signals
coreerrirq[8]	Signal	Master	Core RAS error interrupt
corefaultirq[8]	Signal	Master	Core RAS fault interrupt
cp15sdisable[8]	Signal	Slave	This signal disables write access to some system control processor registers
cpuhalt[8]	Signal	Slave	Raising this signal will put the core into halt mode.
cti[8]	v8EmbeddedCrossTrigger_controlprotocol	Master	-
cti0extin[4]	Signal	Slave	CTI trace inputs for core 0.
cti0extout[4]	Signal	Master	CTI trace outputs for core 0.
cti1extin[4]	Signal	Slave	CTI trace inputs for core 1.
cti1extout[4]	Signal	Master	CTI trace outputs for core 1.
cti2extin[4]	Signal	Slave	CTI trace inputs for core 2.
cti2extout[4]	Signal	Master	CTI trace outputs for core 2.
cti3extin[4]	Signal	Slave	CTI trace inputs for core 3.
cti3extout[4]	Signal	Master	CTI trace outputs for core 3.
cti4extin[4]	Signal	Slave	CTI trace inputs for core 4.
cti4extout[4]	Signal	Master	CTI trace outputs for core 4.
cti5extin[4]	Signal	Slave	CTI trace inputs for core 5.
cti5extout[4]	Signal	Master	CTI trace outputs for core 5.
cti6extin[4]	Signal	Slave	CTI trace inputs for core 6.
cti6extout[4]	Signal	Master	CTI trace outputs for core 6.
cti7extin[4]	Signal	Slave	CTI trace inputs for core 7.
cti7extout[4]	Signal	Master	CTI trace outputs for core 7.
ctidbgirq[8]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	-
dbgnopwrdown[8]	Signal	Master	These signals relate to core power down.

Name	Protocol	Type	Description
dbgpwrdownack[8]	Signal	Master	Debug power down acknowledge.
dbgpwrdownreq[8]	Signal	Slave	Debug power down request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
external_trace_reset[8]	Signal	Slave	ETMv4 External Trace Reset signal.
fiq[8]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[8]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[8]	Signal	Slave	This signal drives the CPUs interrupt handling.
l2reset	Signal	Slave	This signal resets timer and interrupt controller and I2cache
llpp_m[8]	PVBus	Master	LLPP (Low-Latency Peripheral Port).
llram_m	PVBus	Master	LLRAM Port
macp_s	PVBus	Slave	MACP slave interface
memorymapped_debug_s	PVBus	Slave	External debug interface.
periphbase	Value_64	Slave	This port sets the base address of private peripheral region
pmuirq[8]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster irq signal
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wakeup request
ppu_core_irq[8]	Signal	Master	PPU core irq signal
ppu_core_wakerequest[8]	Signal	Slave	PPU core wakeup request
presetdbg[8]	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
reset	Signal	Slave	-
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
rvbar[8]	Value_64	Slave	Reset vector base address.
sei[8]	Signal	Slave	Per core System Error physical pins
spiden	Signal	Slave	Secure invasive debug enable.
spp_m	PVBus	Master	SPP (Shared Peripheral Port).
standbywfe[8]	Signal	Master	This signal indicates if a core is in WFE state
standbywfi[8]	Signal	Master	This signal indicates if a core is in WFI state

Name	Protocol	Type	Description
ticks[8]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trace_unit_reset[8]	Signal	Slave	ETMv4 Trace Unit Reset signal.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[8]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[8]	Signal	Slave	This signal drives the CPUs virtual fast-interrupt handling.
virq[8]	Signal	Slave	This signal drives the CPUs virtual interrupt handling.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).

Parameters for ARM Cortex R82AECT

BROADCASTATOMIC

Enable broadcasting of atomic operation to NC/DEV memory. The broadcastatomic signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTATOMICL

Enable broadcasting of atomic operation to LLRAM memory. The broadcastatomicl signal will override this value and it will be functional for revision 2

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CCSIDR-L1D_override

If nonzero, override the value presented in CCSIDR for L1D (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L1I_override

If nonzero, override the value presented in CCSIDR for L1I (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L2_override

If nonzero, override the value presented in CCSIDR for L2 (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L3_override

If nonzero, allow L3 selection in CSSELR and present this value in CCSIDR (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CFGTFPEN_pin_reset

CFGTFPEN pin at reset

Type

bool

Default value

0x0

CHI

Selects the type of protocol the Main Manager(MM) interface implements. 0, MM port configured as AXI. 1, MM port configured as CHI.

Type

bool

Default value

0x0

CLUSTER_ID

CLUSTER_ID[15:8] equivalent to CFGMPIDRAFF3, CLUSTER_ID[7:0] equivalent to CFGMPIDRAFF2

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x0

NUM_CORES

Number of cores in cluster

Type

int

Default value

0x1

PA_SIZE

Physical address range supported (FEAT_LPA).

Type

int

Default value

0x28

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

VMSA_supported

VMSA is supported at EL1

Type

bool

Default value

0x1

bus_protection_enable_at_reset

Equivalent to CFGBUSPROTEN

Type

bool

Default value

0x0

core_power_on_by_default

Equivalent to PPU_RST_STATE. 0 = Cluster PPU and all core PPUs reset to OFF, 1 = Cluster PPU and all core PPUs reset to ON.

Type

bool

Default value

0x1

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.RVBAR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.TEINIT

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.ase-present

Set whether the model has been built with NEON support

Type

bool

Default value

0x1

cpuX.dcache-size

L1 D-Cache size in bytes

Type

int

Default value

0x10000

cpuX.dtcn_base

Sets the 16K aligned base address of DTCM

Type

int

Default value

0x0

cpuX.dtcn_size

Sets the size of DTCM (in bytes)

Type

int

Default value

0x0

cpuX.dtcn_stretch_clk

Whether DTCM clock stretched to occupy full cycle

Type

bool

Default value

0x0

cpuX.dtcn_wait

DTCM accesses wait states: 0-3 cycles

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.icache-size

L1 I-Cache size in bytes

Type

int

Default value

0x10000

cpuX.itcm_base

Sets the 16K aligned base address of ITCM

Type

int

Default value

0x0

cpuX.itcm_size

Sets the size of ITCM (in bytes)

Type

int

Default value

0x0

cpuX.itcm_stretch_clk

Whether ITCM clock stretched to occupy full cycle

Type

bool

Default value

0x0

cpuX.itcm_wait

ITCM accesses wait states: 0-3 cycles

Type

int

Default value

0x0

cpuX.llpp.base

Equivalent to CFGLLPPBASEADDR

Type

int

Default value

0x0

cpuX.llpp.size

Equivalent to CFGLLPPSIZE

Type

int

Default value

0x8000000

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.tcm.a.enable

Equivalent to CFGITCMENm

Type

bool

Default value

0x0

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-dp-present

Whether double-precision floating point feature is implemented (FEAT_F64MM).

Type

bool

Default value

0x1

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_lock_step

Whether the core is configured in Dual Core Lock Step mode (FEAT_DCLS).

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

gicv3.BPR-min

The minimum value for the GICC_BPR register (non-secure version will be 1 + this value).

Type

int

Default value

0x2

gicv3.VBPR-min

The minimum value for the GICV_BPR register (non-secure version will be 1 + this value).

Type

int

Default value

0x2

has_dense_mem_map

If true, the cluster follows the dense memory map else it implements the sparse memory map.

Type

bool

Default value

0x0

has_impdef_transient_fault_protection

Support the Transient Fault Protection (TFP) flop parity errors through RAS registers (FEAT_TFP)

Type

bool

Default value

0x1

has_llpp

Equivalent to CFGLLPPIMP

Type

bool

Default value

0x1

has_pmc

Programmable MBIST controllers implemented

Type

bool

Default value

0x0

has_spp

Equivalent to CFGSPPIMP

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x400000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory.ext_slave_base

Equivalent to CFGACELSTCMBASEADDR

Type

int

Default value

0x0

memory.has_llram

Equivalent to CFGLLRAMIMP

Type

bool

Default value

0x1

memory.llram_base

Equivalent to CFGLLRAMBASEADDR

Type

int

Default value

0x20000000

memory.llram_enable_at_reset

Equivalent to CFGLLRAMEN

Type

bool

Default value

0x1

memory.llram_shared

Equivalent to CFGLLRAMSHARED and it is only functional for revision 2

Type

bool

Default value

0x0

memory.llram_size

Size of the LLRAM

Type

int

Default value

0x10000000

num_protection_regions_s1

Number of v8-R protection regions

Type

int

Default value

0x10

num_protection_regions_s2

Number of v8-R hyp protection regions

Type

int

Default value

0x10

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

ram_protection_enable_at_reset

Equivalent to CFGGRAMPROTEN

Type

bool

Default value

0x0

spp.base

Equivalent to CFGSPPBASEADDR

Type

int

Default value

0x0

spp.size

Sets the size of SPP(in bytes)

Type

int

Default value

0x8000000

stage12_tlb_size

If VMSA is supported at stage1, number of stage1+2 tlb entries. If instruction_tlb_size !=0, this is treated as dtlb size

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

walk_cache_latency
Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type
int

Default value
0x0

3.5.68 ARMCortexR82CT

ARMCortexR82CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-461: IP revisions support

Revision	Quality level
r0p0	Preliminary support
r1p1	Preliminary support
r2p1	Preliminary support
r3p1	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Table 3-462: Model quality changes

From	To
r0p0=pre	r0p0=pre
r1p1=pre	r1p1=pre
	r2p0=pre
	r3p0=pre

Table 3-463: IP revision changes

From	To
r0p0	r0p0
r1p1	r1p1
	r2p1
	r3p1

Parameters added:

- CHI

- PA_SIZE

Parameters removed:

- bus_type

Iris and MTI instances for ARM CortexR82CT

This model has the following Iris instances:

Table 3-464: ARM CortexR82CT Iris instances

InstanceName	ComponentName
ARM CortexR82CT	Cluster_ARM_Cortex-R82
ARM CortexR82CT.AMU	PVBusLogger
ARM CortexR82CT.AMU.mapper	PVBusMapper
ARM CortexR82CT.DAP	PVBusLogger
ARM CortexR82CT.DAP.mapper	PVBusMapper
ARM CortexR82CT.DSU	DSU
ARM CortexR82CT.DSU.PPU_cluster	PPUv1
ARM CortexR82CT.DSU.PPU_cluster.busslave	PVBusSlave
ARM CortexR82CT.DSU.PPU_core0	PPUv1
ARM CortexR82CT.DSU.PPU_core0.busslave	PVBusSlave
ARM CortexR82CT.DSU.utility_slave[0]	PVBusSlave
ARM CortexR82CT.MMAP	PVBusLogger
ARM CortexR82CT.MMAP.mapper	PVBusMapper
ARM CortexR82CT.cpu0	ARM_Cortex-R82
ARM CortexR82CT.cpu0.UTLB	TLB
ARM CortexR82CT.cpu0.dtlb	TlbCadi
ARM CortexR82CT.cpu0.gicv3_cpu_if	GICv3CPUInterface
ARM CortexR82CT.cpu0.l1dcache	PVCache
ARM CortexR82CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM CortexR82CT.cpu0.l1icache	PVCache
ARM CortexR82CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM CortexR82CT.ext_bus	PVBusLogger
ARM CortexR82CT.ext_bus.mapper	PVBusMapper
ARM CortexR82CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARM CortexR82CT.l2_cache	PVCache
ARM CortexR82CT.l2_cache.upstream[0]	PVBusSlave
ARM CortexR82CT.l2_cache.upstream[10]	PVBusSlave
ARM CortexR82CT.l2_cache.upstream[11]	PVBusSlave
ARM CortexR82CT.l2_cache.upstream[12]	PVBusSlave
ARM CortexR82CT.l2_cache.upstream[13]	PVBusSlave
ARM CortexR82CT.l2_cache.upstream[14]	PVBusSlave

InstanceName	ComponentName
ARMCortexR82CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexR82CT.llram_atomic_filter	LLRAMAtomicOpFilter
ARMCortexR82CT.llram_atomic_filter.atomic_bus_slave	PVBusSlave
ARMCortexR82CT.llram_atomic_filter.filter	PVBusMapper
ARMCortexR82CT.llram_atomic_filter.llram_exclusive_monitor	PVBusExclusiveMonitor
ARMCortexR82CT.llram_atomic_filter.llram_exclusive_monitor.bus_mapper	PVBusMapper
ARMCortexR82CT.llram_atomic_filter.pvbusmaster	PVBusMaster
ARMCortexR82CT.llram_coherent_interconnect	PVCache
ARMCortexR82CT.llram_coherent_interconnect.upstream[0]	PVBusSlave
ARMCortexR82CT.llram_coherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR82CT.llram_coherent_interconnect.upstream[2]	PVBusSlave

This model has the following MTI trace components:

Table 3-465: ARMCortexR82CT MTI instances

InstanceName	ComponentName
ARMCortexR82CT	ARMv8Cluster
ARMCortexR82CT.AMU	PVBusLogger
ARMCortexR82CT.AMU.mapper	PVBusMapper
ARMCortexR82CT.DAP	PVBusLogger
ARMCortexR82CT.DAP.mapper	PVBusMapper
ARMCortexR82CT.DSU	DSU
ARMCortexR82CT.DSU.PPU_cluster	PPUv1
ARMCortexR82CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexR82CT.DSU.PPU_core0	PPUv1
ARMCortexR82CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexR82CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexR82CT.MMAP	PVBusLogger
ARMCortexR82CT.MMAP.mapper	PVBusMapper
ARMCortexR82CT.cpu0	ARM_Cortex-R82

InstanceName	ComponentName
ARMCortexR82CT.cpu0.UTLB	TLB
ARMCortexR82CT.cpu0.gicv3_cpu_if	GICv3CPUInterface
ARMCortexR82CT.cpu0.l1dcache	PVCache
ARMCortexR82CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexR82CT.cpu0.l1icache	PVCache
ARMCortexR82CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexR82CT.ext_bus	PVBusLogger
ARMCortexR82CT.ext_bus.mapper	PVBusMapper
ARMCortexR82CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder
ARMCortexR82CT.l2_cache	PVCache
ARMCortexR82CT.l2_cache.upstream[0]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[10]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[11]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[12]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[13]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[14]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[15]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[16]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[1]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[2]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[3]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[4]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[5]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[6]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[7]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[8]	PVBusSlave
ARMCortexR82CT.l2_cache.upstream[9]	PVBusSlave
ARMCortexR82CT.llram_atomic_filter	LLRAMAtomicOpFilter
ARMCortexR82CT.llram_atomic_filter.atomic_bus_slave	PVBusSlave
ARMCortexR82CT.llram_atomic_filter.filter	PVBusMapper
ARMCortexR82CT.llram_atomic_filter.llram_exclusive_monitor	PVBusExclusiveMonitor
ARMCortexR82CT.llram_atomic_filter.llram_exclusive_monitor.bus_mapper	PVBusMapper
ARMCortexR82CT.llram_atomic_filter.pvbusmaster	PVBusMaster
ARMCortexR82CT.llram_coherent_interconnect	PVCache
ARMCortexR82CT.llram_coherent_interconnect.upstream[0]	PVBusSlave
ARMCortexR82CT.llram_coherent_interconnect.upstream[1]	PVBusSlave
ARMCortexR82CT.llram_coherent_interconnect.upstream[2]	PVBusSlave

ARMCortexR82CT contains the following CADI targets:

- ARM_Cortex-R82
- Cluster_ARM_Cortex-R82
- PVCache
- TlbCadi

About ARMCortexR82CT

To simulate the r1p0 model, use the following parameters:

- revision_number=1
- VMSA_supported=1

The following models also exist:

- ARMCortexR82x1CT
- ARMCortexR82x2CT
- ARMCortexR82x3CT
- ARMCortexR82x4CT
- ARMCortexR82x6CT
- ARMCortexR82x8CT

Limitations

Dense memory map support has been added for the Utility bus only.

Ports for ARMCortexR82CT

Table 3-466: Ports

Name	Protocol	Type	Description
acel_s	PVBus	Slave	External Slave port. Equivalent to AXIS port.
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastatomicl	Signal	Slave	BROADCASTATOMIC pin for LLRAM
broadcastcachemaint	Signal	Slave	Enable broadcasting of cache maintenance operations to downstream caches.
broadcastouter	Signal	Slave	Enable broadcasting of Outer Shareable transactions.
cache_validation_control	Value	Slave	This signal provides default exception handling state.
cfgendianess[8]	Signal	Slave	This signal if for EE bit initialisation
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
cluster_pcsn_pchannel	PChannel	Master	Cluster PCSM signal
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.

Name	Protocol	Type	Description
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPSIRQ[8]	Signal	Master	Timer signals to SOC
CNTHVSIRQ[8]	Signal	Master	Timer signals to SOC
CNTPSIRQ[8]	Signal	Master	Timer signals to SOC
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module
CNTVIRQ[8]	Signal	Master	Timer signals to SOC
commirq[8]	Signal	Master	Interrupt signal from debug communication channel.
core_pcs m_pchannel[8]	PChannel	Master	Core PCSM signals
coreerrirq[8]	Signal	Master	Core RAS error interrupt
corefaultirq[8]	Signal	Master	Core RAS fault interrupt
cp15sdisable[8]	Signal	Slave	This signal disables write access to some system control processor registers
cpuhalt[8]	Signal	Slave	Raising this signal will put the core into halt mode.
cti[8]	v8EmbeddedCrossTrigger_controlprotocol	Master	-
cti0extin[4]	Signal	Slave	CTI trace inputs for core 0.
cti0extout[4]	Signal	Master	CTI trace outputs for core 0.
cti1extin[4]	Signal	Slave	CTI trace inputs for core 1.
cti1extout[4]	Signal	Master	CTI trace outputs for core 1.
cti2extin[4]	Signal	Slave	CTI trace inputs for core 2.
cti2extout[4]	Signal	Master	CTI trace outputs for core 2.
cti3extin[4]	Signal	Slave	CTI trace inputs for core 3.
cti3extout[4]	Signal	Master	CTI trace outputs for core 3.
cti4extin[4]	Signal	Slave	CTI trace inputs for core 4.
cti4extout[4]	Signal	Master	CTI trace outputs for core 4.
cti5extin[4]	Signal	Slave	CTI trace inputs for core 5.
cti5extout[4]	Signal	Master	CTI trace outputs for core 5.
cti6extin[4]	Signal	Slave	CTI trace inputs for core 6.
cti6extout[4]	Signal	Master	CTI trace outputs for core 6.
cti7extin[4]	Signal	Slave	CTI trace inputs for core 7.
cti7extout[4]	Signal	Master	CTI trace outputs for core 7.
ctidbgirq[8]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	-
dbgnopwrdown[8]	Signal	Master	These signals relate to core power down.
dbgpwrdownack[8]	Signal	Master	Debug power down acknowledge.
dbgpwrdownreq[8]	Signal	Slave	Debug power down request.
dev_debug_s	PVBus	Slave	External debug interface.

Name	Protocol	Type	Description
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
external_trace_reset[8]	Signal	Slave	ETMv4 External Trace Reset signal.
fiq[8]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[8]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[8]	Signal	Slave	This signal drives the CPUs interrupt handling.
l2reset	Signal	Slave	This signal resets timer and interrupt controller and I2cache
llpp_m[8]	PVBus	Master	LLPP (Low-Latency Peripheral Port).
llram_m	PVBus	Master	LLRAM Port
macp_s	PVBus	Slave	MACP slave interface
memorymapped_debug_s	PVBus	Slave	External debug interface.
periphbase	Value_64	Slave	This port sets the base address of private peripheral region
pmuirq[8]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster irq signal
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wakeup request
ppu_core_irq[8]	Signal	Master	PPU core irq signal
ppu_core_wakerequest[8]	Signal	Slave	PPU core wakeup request
presetdbg[8]	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
reset	Signal	Slave	-
romaddr	Value_64	Slave	Debug ROM base address.
romaddrv	Signal	Slave	Debug ROM base address valid.
rvbar[8]	Value_64	Slave	Reset vector base address.
sei[8]	Signal	Slave	Per core System Error physical pins
spiden	Signal	Slave	Secure invasive debug enable.
spp_m	PVBus	Master	SPP (Shared Peripheral Port).
standbywfe[8]	Signal	Master	This signal indicates if a core is in WFE state
standbywfi[8]	Signal	Master	This signal indicates if a core is in WFI state
ticks[8]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.

Name	Protocol	Type	Description
trace_unit_reset[8]	Signal	Slave	ETMv4 Trace Unit Reset signal.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[8]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[8]	Signal	Slave	This signal drives the CPUs virtual fast-interrupt handling.
virq[8]	Signal	Slave	This signal drives the CPUs virtual interrupt handling.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).

Parameters for ARM CortexR82CT

BROADCASTATOMIC

Enable broadcasting of atomic operation to NC/DEV memory. The broadcastatomic signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTATOMICL

Enable broadcasting of atomic operation to LLRAM memory. The broadcastatomicl signal will override this value and it will be functional for revision 2

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

CCSIDR-L1D_override

If nonzero, override the value presented in CCSIDR for L1D (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L1I_override

If nonzero, override the value presented in CCSIDR for L1I (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L2_override

If nonzero, override the value presented in CCSIDR for L2 (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CCSIDR-L3_override

If nonzero, allow L3 selection in CSSELR and present this value in CCSIDR (this is cosmetic and does not affect cache behaviour).

Type

int

Default value

0x0

CHI

Selects the type of protocol the Main Manager(MM) interface implements. 0, MM port configured as AXI. 1, MM port configured as CHI.

Type

bool

Default value

0x0

CLUSTER_ID

CLUSTER_ID[15:8] equivalent to CFGMPIDRAFF3, CLUSTER_ID[7:0] equivalent to CFGMPIDRAFF2

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

DBGROMADDR

Initialization value of DBGDRAR register. Bits[55:12] of this register specify the ROM table physical address.

Type

int

Default value

0x0

DBGROMADDRV

If true, set bits[1:0] of the CP15 DBGDRAR to indicate that the address is valid

Type

bool

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x0

NUM_CORES

Number of cores in cluster

Type

int

Default value

0x1

PA_SIZE

Physical address range supported (FEAT_LPA).

Type

int

Default value

0x28

PERIPHBASE

Base address of peripheral memory space

Type

int

Default value

0x13080000

VMSA_supported

VMSA is supported at EL1

Type

bool

Default value

0x0

core_power_on_by_default

Equivalent to PPU_RST_STATE. 0 = Cluster PPU and all core PPUs reset to OFF, 1 = Cluster PPU and all core PPUs reset to ON.

Type

bool

Default value

0x1

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CP15SDISABLE

Initialize to disable access to some CP15 registers

Type

bool

Default value

0x0

cpuX.RVBAR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.TEINIT

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.ase-present

Set whether the model has been built with NEON support

Type

bool

Default value

0x1

cpuX.dcache-size

L1 D-Cache size in bytes

Type

int

Default value

0x10000

cpuX.dtcn_base

Sets the 16K aligned base address of DTCM

Type

int

Default value

0x0

cpuX.dtcn_size

Sets the size of DTCM (in bytes)

Type

int

Default value

0x0

cpuX.dtcn_stretch_clk

Whether DTCM clock stretched to occupy full cycle

Type

bool

Default value

0x0

cpuX.dtcn_wait

DTCM accesses wait states: 0-3 cycles

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.icache-size

L1 I-Cache size in bytes

Type

int

Default value

0x10000

cpuX.itcm_base

Sets the 16K aligned base address of ITCM

Type

int

Default value

0x0

cpuX.itcm_size

Sets the size of ITCM (in bytes)

Type

int

Default value

0x0

cpuX.itcm_stretch_clk

Whether ITCM clock stretched to occupy full cycle

Type

bool

Default value

0x0

cpuX.itcm_wait

ITCM accesses wait states: 0-3 cycles

Type

int

Default value

0x0

cpuX.llpp.base

Equivalent to CFGLLPPBASEADDR

Type

int

Default value

0x0

cpuX.llpp.size

Equivalent to CFGLLPPSIZE

Type

int

Default value

0x8000000

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.tcm.a.enable

Equivalent to CFGITCMENm

Type

bool

Default value

0x0

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-dp-present

Whether double-precision floating point feature is implemented (FEAT_F64MM).

Type

bool

Default value

0x1

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

enable_lock_step

Whether the core is configured in Dual Core Lock Step mode (FEAT_DCLS).

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

gicv3.BPR-min

The minimum value for the GICC_BPR register (non-secure version will be 1 + this value).

Type

int

Default value

0x2

gicv3.VBPR-min

The minimum value for the GICV_BPR register (non-secure version will be 1 + this value).

Type

int

Default value

0x2

has_dense_mem_map

If true, the cluster follows the dense memory map else it implements the sparse memory map.

Type

bool

Default value

0x0

has_llpp

Equivalent to CFGLLPPIMP

Type

bool

Default value

0x1

has_spp

Equivalent to CFGSPPIMP

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or

miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

`icache-read_latency`

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

`icache-state_modelled`

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

`l2cache-hit_latency`

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l2cache-maintenance_latency`

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x400000

l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory.ext_slave_base

Equivalent to CFGACELSTCMBASEADDR

Type

int

Default value

0x0

memory.has_llram

Equivalent to CFGLLRAMIMP

Type

bool

Default value

0x1

memory.llram_base

Equivalent to CFGLLRAMBASEADDR

Type

int

Default value

0x20000000

memory.llram_enable_at_reset

Equivalent to CFGLLRAMEN

Type

bool

Default value

0x1

memory.llram_shared

Equivalent to CFGLLRAMSHARED and it is only functional for revision 2

Type

bool

Default value

0x0

memory.llram_size

Size of the LLRAM

Type

int

Default value

0x10000000

num_protection_regions_s1

Number of v8-R protection regions

Type

int

Default value

0x10

num_protection_regions_s2

Number of v8-R hyp protection regions

Type

int

Default value

0x10

patch_level

Patch level of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Revision field in MIDR/MIDR_EL1. Corresponds to the patch number Y in rXpY.

Type

int

Default value

0x1

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

ram_protection_enable_at_reset

Equivalent to CFGGRAMPROTEN

Type

bool

Default value

0x0

revision_number

Revision number of TRM implemented by the model. Changing the value can change the model behaviour. Visible as the Variant field in MIDR/MIDR_EL1. Corresponds to the revision number X in rXpY.

Type

int

Default value

0x1

spp.base

Equivalent to CFGSPPBASEADDR

Type

int

Default value

0x0

spp.size

Sets the size of SPP(in bytes)

Type

int

Default value

0x8000000

stage12_tlb_size

If VMSA is supported at stage1, number of stage1+2 tlb entries. If instruction_tlb_size !=0, this is treated as dtlb size

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.69 **ARMCortexX1CCT**

ARMCortexX1CCT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-467: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- tlbi_stall_enabled

Iris and MTI instances for ARM CortexX1CCT

This model has the following Iris instances:

Table 3-468: ARM CortexX1CCT Iris instances

InstanceName	ComponentName
ARM CortexX1CCT	Cluster_ARM_Cortex-X1C
ARM CortexX1CCT.AMU	PVBusLogger
ARM CortexX1CCT.AMU.mapper	PVBusMapper
ARM CortexX1CCT.DAP	PVBusLogger
ARM CortexX1CCT.DAP.mapper	PVBusMapper
ARM CortexX1CCT.DSU	DSU
ARM CortexX1CCT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM CortexX1CCT.DSU.mpam_busslave	PVBusSlave
ARM CortexX1CCT.DSU.shared_cache	PVCache
ARM CortexX1CCT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM CortexX1CCT.DSU.shared_cache.upstream[1]	PVBusSlave
ARM CortexX1CCT.DSU.shared_cache.upstream[2]	PVBusSlave
ARM CortexX1CCT.DSU.shared_cache.upstream[3]	PVBusSlave
ARM CortexX1CCT.MMAP	PVBusLogger
ARM CortexX1CCT.MMAP.mapper	PVBusMapper
ARM CortexX1CCT.cpu0	ARM_Cortex-X1C
ARM CortexX1CCT.cpu0.UTLB	TLB
ARM CortexX1CCT.cpu0.dtlb	TlbCadi
ARM CortexX1CCT.cpu0.l1dcache	PVCache
ARM CortexX1CCT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM CortexX1CCT.cpu0.l1icache	PVCache
ARM CortexX1CCT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM CortexX1CCT.cpu0.l2cache	PVCache
ARM CortexX1CCT.cpu0.l2cache.upstream[0]	PVBusSlave
ARM CortexX1CCT.cpu0.l2cache.upstream[1]	PVBusSlave
ARM CortexX1CCT.ext_bus	PVBusLogger
ARM CortexX1CCT.ext_bus.mapper	PVBusMapper
ARM CortexX1CCT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-469: ARM CortexX1CCT MTI instances

InstanceName	ComponentName
ARM CortexX1CCT	ARMv8Cluster
ARM CortexX1CCT.AMU	PVBusLogger
ARM CortexX1CCT.AMU.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexX1CCT.DAP	PVBusLogger
ARMCortexX1CCT.DAP.mapper	PVBusMapper
ARMCortexX1CCT.DSU	DSU
ARMCortexX1CCT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexX1CCT.DSU.mpam_busslave	PVBusSlave
ARMCortexX1CCT.DSU.shared_cache	PVCache
ARMCortexX1CCT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexX1CCT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexX1CCT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexX1CCT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexX1CCT.MMAP	PVBusLogger
ARMCortexX1CCT.MMAP.mapper	PVBusMapper
ARMCortexX1CCT.cpu0	ARM_Cortex-X1C
ARMCortexX1CCT.cpu0.UTLB	TLB
ARMCortexX1CCT.cpu0.l1dcache	PVCache
ARMCortexX1CCT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexX1CCT.cpu0.l1icache	PVCache
ARMCortexX1CCT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexX1CCT.cpu0.l2cache	PVCache
ARMCortexX1CCT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexX1CCT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexX1CCT.ext_bus	PVBusLogger
ARMCortexX1CCT.ext_bus.mapper	PVBusMapper
ARMCortexX1CCT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexX1CCT contains the following CADI targets:

- ARM_Cortex-X1C
- Cluster_ARM_Cortex-X1C
- PVCache
- TlbCadi

About ARMCortexX1CCT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.

- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Cache stashing capability.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.



The `cfgsdisable` signal will be removed in a future release.

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexX1CCT

Table 3-470: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
AENDMP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTARTMP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.

Name	Protocol	Type	Description
cfgend[8]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[8]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[8]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[8]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[8]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[8]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[8]	Signal	Master	Timer signals to SOC.
commirq[8]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[8]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cpuporeset[8]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[8]	Signal	Slave	Disable cryptography extensions after reset.
cti[8]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[8]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[8]	Signal	Master	No power-down request.
dbgpwrupreq[8]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[8]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[8]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[8]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.

Name	Protocol	Type	Description
pchannel_core[8]	PChannel	Slave	PChannels for cores
pmbirq[8]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[8]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[8]	Signal	Slave	Per core RAM Error Interrupt.
reset[8]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[8]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[8]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[8]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[8]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[8]	Signal	Slave	Virtualised FIQ.
vinithi[8]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[8]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[8]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexX1CCT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x0

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x100000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.70 **ARMCortexX1CT**

ARMCortexX1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-471: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- tlbi_stall_enabled

Iris and MTI instances for ARM CortexX1CT

This model has the following Iris instances:

Table 3-472: ARM CortexX1CT Iris instances

InstanceName	ComponentName
ARM CortexX1CT	Cluster_ARM_Cortex-X1
ARM CortexX1CT.AMU	PVBusLogger
ARM CortexX1CT.AMU.mapper	PVBusMapper
ARM CortexX1CT.DAP	PVBusLogger
ARM CortexX1CT.DAP.mapper	PVBusMapper
ARM CortexX1CT.DSU	DSU
ARM CortexX1CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARM CortexX1CT.DSU.mpam_busslave	PVBusSlave
ARM CortexX1CT.DSU.shared_cache	PVCache
ARM CortexX1CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM CortexX1CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARM CortexX1CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARM CortexX1CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARM CortexX1CT.MMAP	PVBusLogger
ARM CortexX1CT.MMAP.mapper	PVBusMapper
ARM CortexX1CT.cpu0	ARM_Cortex-X1
ARM CortexX1CT.cpu0.UTLB	TLB
ARM CortexX1CT.cpu0.dtlb	TlbCadi
ARM CortexX1CT.cpu0.l1dcache	PVCache
ARM CortexX1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM CortexX1CT.cpu0.l1icache	PVCache
ARM CortexX1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM CortexX1CT.cpu0.l2cache	PVCache
ARM CortexX1CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARM CortexX1CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARM CortexX1CT.ext_bus	PVBusLogger
ARM CortexX1CT.ext_bus.mapper	PVBusMapper
ARM CortexX1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-473: ARM CortexX1CT MTI instances

InstanceName	ComponentName
ARM CortexX1CT	ARMv8Cluster
ARM CortexX1CT.AMU	PVBusLogger
ARM CortexX1CT.AMU.mapper	PVBusMapper

InstanceName	ComponentName
ARMCortexX1CT.DAP	PVBusLogger
ARMCortexX1CT.DAP.mapper	PVBusMapper
ARMCortexX1CT.DSU	DSU
ARMCortexX1CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexX1CT.DSU.mpam_busslave	PVBusSlave
ARMCortexX1CT.DSU.shared_cache	PVCache
ARMCortexX1CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexX1CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexX1CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexX1CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexX1CT.MMAP	PVBusLogger
ARMCortexX1CT.MMAP.mapper	PVBusMapper
ARMCortexX1CT.cpu0	ARM_Cortex-X1
ARMCortexX1CT.cpu0.UTLB	TLB
ARMCortexX1CT.cpu0.l1dcache	PVCache
ARMCortexX1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexX1CT.cpu0.l1icache	PVCache
ARMCortexX1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexX1CT.cpu0.l2cache	PVCache
ARMCortexX1CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexX1CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexX1CT.ext_bus	PVBusLogger
ARMCortexX1CT.ext_bus.mapper	PVBusMapper
ARMCortexX1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexX1CT contains the following CADI targets:

- ARM_Cortex-X1
- Cluster_ARM_Cortex-X1
- PVCache
- TlbCadi

About ARMCortexX1CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.

- Per-core clock.
- `BROADCASTPERSIST` pin.

The model does not support the following features:

- `BROADCASTCACHEMAINTPOU` pin.
- `COREINSTRRET`, `COREINSTRRUN`, or `nPMBIRQ` signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals `DBGPWRDUP` and `DBGIRSTREQ` are not implemented, but `DBGPWRUPREQ` and `DBGNOPWRDWN` are implemented.
- Cache stashing capability.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.

This model has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following models also exist, with a fixed number of cores per cluster:

- ARMCortexX1x1CT.
- ARMCortexX1x2CT.
- ARMCortexX1x3CT.
- ARMCortexX1x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexX1CT

Table 3-474: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AENDMP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).

Name	Protocol	Type	Description
ASTARTMP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[4]	Signal	Slave	This signal if for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[4]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[4]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Timer signals to SOC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[4]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cpuporeset[4]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.

Name	Protocol	Type	Description
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[4]	PChannel	Slave	PChannels for cores
pmbirq[4]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[4]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexX1CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size `dcache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x0

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x100000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.71 ARMCortexX2CT

ARMCortexX2CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-475: IP revisions support

Revision	Quality level
r2p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- mpmm_accumulator_multiplier
- tlbi_stall_enabled

Iris and MTI instances for ARM CortexX2CT

This model has the following Iris instances:

Table 3-476: ARM CortexX2CT Iris instances

InstanceName	ComponentName
ARM CortexX2CT	Cluster_ARM_Cortex-X2
ARM CortexX2CT.AMU	PVBusLogger
ARM CortexX2CT.AMU.mapper	PVBusMapper
ARM CortexX2CT.DAP	PVBusLogger
ARM CortexX2CT.DAP.mapper	PVBusMapper
ARM CortexX2CT.DSU	DSU
ARM CortexX2CT.DSU.PPU_cluster	PPUv1
ARM CortexX2CT.DSU.PPU_cluster.busslave	PVBusSlave
ARM CortexX2CT.DSU.PPU_core0	PPUv1
ARM CortexX2CT.DSU.PPU_core0.busslave	PVBusSlave
ARM CortexX2CT.DSU.mpam_busslave	PVBusSlave
ARM CortexX2CT.DSU.shared_cache	PVCache
ARM CortexX2CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARM CortexX2CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARM CortexX2CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARM CortexX2CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARM CortexX2CT.DSU.utility_slave[0]	PVBusSlave
ARM CortexX2CT.MMAP	PVBusLogger
ARM CortexX2CT.MMAP.mapper	PVBusMapper
ARM CortexX2CT.cpu0	ARM_Cortex-X2
ARM CortexX2CT.cpu0.UTLB	TLB
ARM CortexX2CT.cpu0.dtlb	TlbCadi
ARM CortexX2CT.cpu0.l1dcache	PVCache
ARM CortexX2CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARM CortexX2CT.cpu0.l1icache	PVCache
ARM CortexX2CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARM CortexX2CT.cpu0.l2cache	PVCache
ARM CortexX2CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARM CortexX2CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARM CortexX2CT.ext_bus	PVBusLogger
ARM CortexX2CT.ext_bus.mapper	PVBusMapper
ARM CortexX2CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-477: ARMCortexX2CT MTI instances

InstanceName	ComponentName
ARMCortexX2CT	ARMv8Cluster
ARMCortexX2CT.AMU	PVBusLogger
ARMCortexX2CT.AMU.mapper	PVBusMapper
ARMCortexX2CT.DAP	PVBusLogger
ARMCortexX2CT.DAP.mapper	PVBusMapper
ARMCortexX2CT.DSU	DSU
ARMCortexX2CT.DSU.PPU_cluster	PPUv1
ARMCortexX2CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexX2CT.DSU.PPU_core0	PPUv1
ARMCortexX2CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexX2CT.DSU.mpam_busslave	PVBusSlave
ARMCortexX2CT.DSU.shared_cache	PVCache
ARMCortexX2CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexX2CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexX2CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexX2CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexX2CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexX2CT.MMAP	PVBusLogger
ARMCortexX2CT.MMAP.mapper	PVBusMapper
ARMCortexX2CT.cpu0	ARM_Cortex-X2
ARMCortexX2CT.cpu0.UTLB	TLB
ARMCortexX2CT.cpu0.l1dcache	PVCache
ARMCortexX2CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexX2CT.cpu0.l1icache	PVCache
ARMCortexX2CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexX2CT.cpu0.l2cache	PVCache
ARMCortexX2CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexX2CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexX2CT.ext_bus	PVBusLogger
ARMCortexX2CT.ext_bus.mapper	PVBusMapper
ARMCortexX2CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexX2CT contains the following CADI targets:

- ARM_Cortex-X2
- Cluster_ARM_Cortex-X2
- PVCache
- TlbCadi

About ARMCortexX2CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

Support for the following features is planned for a future release:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, and nPMBIRQ signals.
- Transporting architectural metadata tags over TLM2 (OSCI) and AMBA-PV (TLM2 extensions from Arm) bus protocols.

The following features will not be implemented:

- 256-bit wide output transactions.
- Error correction/detection features.
- Self-test features (MBIST).
- Snoop filtering.

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (crypto.dll or crypto.so) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMCortexX2CT

Table 3-478: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).

Name	Protocol	Type	Description
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[12]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[12]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[12]	Signal	Master	Timer signals to SOC
CNTHVIRQ[12]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[12]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[12]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[12]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[12]	Signal	Master	Timer signals to SOC.
commirq[12]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[12]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[12]	PChannel	Master	Core PCSM signals
core_powerdown_out[12]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[12]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency

Name	Protocol	Type	Description
corefaultirq[12]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[12]	Signal	Slave	Disable cryptography extensions after reset.
cti[12]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[12]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[12]	Signal	Master	No power-down request.
dbgpwrupreq[12]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[12]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[12]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[12]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmuirq[12]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[12]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[12]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[12]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[12]	Value_64	Slave	Reset vector base address.

Name	Protocol	Type	Description
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[12]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[12]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[12]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[12]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[12]	Signal	Slave	Virtualised FIQ.
virq[12]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[12]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexX2CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpu_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-read_latency`

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-size`

L2 Cache size in bytes.

Type

int

Default value

0x80000

`cpuX.l2cache-snoop_data_transfer_latency`

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-snoop_issue_latency`

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x20

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x2

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x0

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).

Type

int

Default value

0x2

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive (corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.72 ARMCortexX3CT

ARMCortexX3CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-479: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `mpmm_accumulator_multiplier`
- `tlbi_stall_enabled`

Iris and MTI instances for ARMCortexX3CT

This model has the following Iris instances:

Table 3-480: ARMCortexX3CT Iris instances

InstanceName	ComponentName
ARMCortexX3CT	Cluster_ARM_Cortex-X3
ARMCortexX3CT.AMU	PVBusLogger
ARMCortexX3CT.AMU.mapper	PVBusMapper
ARMCortexX3CT.DAP	PVBusLogger
ARMCortexX3CT.DAP.mapper	PVBusMapper
ARMCortexX3CT.DSU	DSU
ARMCortexX3CT.DSU.PPU_cluster	PPUv1
ARMCortexX3CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexX3CT.DSU.PPU_core0	PPUv1
ARMCortexX3CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexX3CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexX3CT.DSU.mpam_busslave	PVBusSlave
ARMCortexX3CT.DSU.shared_cache	PVCache
ARMCortexX3CT.DSU.shared_cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMCortexX3CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexX3CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexX3CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexX3CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexX3CT.MMAP	PVBusLogger
ARMCortexX3CT.MMAP.mapper	PVBusMapper
ARMCortexX3CT.cpu0	ARM_Cortex-X3
ARMCortexX3CT.cpu0.UTLB	TLB
ARMCortexX3CT.cpu0.dtlb	TlbCadi
ARMCortexX3CT.cpu0.l1dcache	PVCache
ARMCortexX3CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexX3CT.cpu0.l1icache	PVCache
ARMCortexX3CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexX3CT.cpu0.l2cache	PVCache
ARMCortexX3CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexX3CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexX3CT.ext_bus	PVBusLogger
ARMCortexX3CT.ext_bus.mapper	PVBusMapper
ARMCortexX3CT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-481: ARMCortexX3CT MTI instances

InstanceName	ComponentName
ARMCortexX3CT	ARMv8Cluster
ARMCortexX3CT.AMU	PVBusLogger
ARMCortexX3CT.AMU.mapper	PVBusMapper
ARMCortexX3CT.DAP	PVBusLogger
ARMCortexX3CT.DAP.mapper	PVBusMapper
ARMCortexX3CT.DSU	DSU
ARMCortexX3CT.DSU.PPU_cluster	PPUv1
ARMCortexX3CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexX3CT.DSU.PPU_core0	PPUv1
ARMCortexX3CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexX3CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexX3CT.DSU.mpam_busslave	PVBusSlave
ARMCortexX3CT.DSU.shared_cache	PVCache
ARMCortexX3CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexX3CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexX3CT.DSU.shared_cache.upstream[2]	PVBusSlave

InstanceName	ComponentName
ARMCortexX3CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexX3CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexX3CT.MMAP	PVBusLogger
ARMCortexX3CT.MMAP.mapper	PVBusMapper
ARMCortexX3CT.cpu0	ARM_Cortex-X3
ARMCortexX3CT.cpu0.UTLB	TLB
ARMCortexX3CT.cpu0.l1dcache	PVCache
ARMCortexX3CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexX3CT.cpu0.l1icache	PVCache
ARMCortexX3CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexX3CT.cpu0.l2cache	PVCache
ARMCortexX3CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexX3CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexX3CT.ext_bus	PVBusLogger
ARMCortexX3CT.ext_bus.mapper	PVBusMapper
ARMCortexX3CT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder

ARMCortexX3CT contains the following CADI targets:

- ARM_Cortex-X3
- Cluster_ARM_Cortex-X3
- PVCache
- TlbCadi

About ARMCortexX3CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- DynamIQ r3p0.
- L2 cache is supported at the per-core level only and there is no implementation of Core-Complex with shared L2 cache yet.
- P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.
- Utility bus.
- Support for Interrupt signals from SPE is added as `pmbirq[8]`.

Support for the following features is planned for a future release:

- TRBE.
- Transporting architectural metadata tags over TLM2 (OSCI) and AMBA-PV (TLM2 extensions from Arm) bus protocols.
- Core-Complex.
- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.



The `cfgsdisable` signal will be removed in a future release.

The following features will not be implemented:

- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Cache stashing capability.
- Embedded Logic Analyzer (ELA).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Limitations

- Some of the SVE instructions do not raise undefined exceptions when SP is used as a source register.

Ports for ARMCortexX3CT

Table 3-482: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.

Name	Protocol	Type	Description
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[12]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcsmp_channel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[12]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[12]	Signal	Master	Timer signals to SOC
CNTHVIRQ[12]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[12]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[12]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[12]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[12]	Signal	Master	Timer signals to SOC.

Name	Protocol	Type	Description
commirq[12]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[12]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[12]	PChannel	Master	Core PCSM signals
core_powerdown_out[12]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[12]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[12]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[12]	Signal	Slave	Disable cryptography extensions after reset.
cti[12]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[12]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[12]	Signal	Master	No power-down request.
dbgpwrupreq[12]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[12]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[12]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[12]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[12]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[12]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[12]	Signal	Master	PPU core interrupt.

Name	Protocol	Type	Description
ppu_core_wakerequest[12]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[12]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[12]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[12]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[12]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[12]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[12]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[12]	Signal	Slave	Virtualised FIQ.
virq[12]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[12]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM CortexX3CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x100000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

error_record_feature_register

RAS feature register values. An array of JSON objects. The JSON schema for the array is: [{"ED":0x0, "IMPDEF_3_2":0x0, "UI":0x0, "FI":0x0, "UE":0x0, "CFI":0x0, "CEC":0x0, "RP":0x0, "DUI":0x0, "CEO":0x0, "CI":0x0, "TS":0x0, "INJ":0x0, "FRX":0x0, "UC":0x0, "UEU":0x0, "UER":0x0, "UEO":0x0, "DE":0x0, "CE":0x0, "Visibility":"Core"},other_feature_register_values]. Where ED,UI,FI,CE and UE have valid values between 0x0 - 0x3. CFI and DUI have valid values 0x0, 0x2 and 0x3. CEC has valid values 0x0,0x2 or 0x4. RP,CEO,INJ,FRX,UC,UEU,UER,UEO,DE has valid values 0x0 or 0x1. CI and TS has valid values of 0x0, 0x1 and 0x2. Visibility has valid values "Core" or "Cluster"

Type

string

Default value

```
[{"ED":0x2,"IMPDEF_3_2":0x0,"UI":0x2,"FI":0x2,"UE":0x1,"CFI":0x2,"CEC":0x2,"RP":0x1,"DUI":0x0,"CEC":0x2,"Visibility":"Cluster"},
{"ED":0x2,"IMPDEF_3_2":0x0,"UI":0x2,"FI":0x2,"UE":0x1,"CFI":0x2,"CEC":0x2,"RP":0x1,"DUI":0x0,"CEC":0x2,"Visibility":"Cluster"}]
```

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x20

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

ext_abort_normal_noncacheable_read_is_sync

Synchronous reporting of normal noncacheable-read external aborts

Type

bool

Default value

0x1

ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x2

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

l3cache-has_mpam

L3 Cache has MPAM support

Type

bool

Default value

0x1

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit

or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-read_latency`

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l3cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-size`

L3 Cache size in bytes.

Type

int

Default value

0x40000

`l3cache-snoop_data_transfer_latency`

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-snoop_issue_latency`

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-ways`

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

pseudo_fault_generation_feature_register

ARMv8.4 Standard Pseudo-fault generation feature register values. JSON schema for the parameter value is: [{"OF":false, "UC":false, "UEU":false, "UER":false, "UEO":false, "DE":false, "CE":0x0, "CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":false, "NA":false}, other_pseudo-fault_generating_features_register_values]. Where OF, UC, UEU, UER, UEO, DE, CI, ER, PN, AV, MV, SYN, and R have valid false(NOT_SUPPORTED) and true(FEATURE_CONTROLLABLE), where CE can have 0(NOT_SUPPORTED), 1(NONSPECIFIC_CE_SUPPORTED) and 3(TRANSIENT_OR_PERSISTENT_CE_SUPPORTED) and NA can have false(component fakes detection on next access) or true(component fakes detection spontaneously). Effective only when ERXFR's INJ field allows it or has_ras_fault_injection is true.

Type

string

Default value

```
[{"OF":true, "UC":true, "UEU":false, "UER":false, "UEO":false, "DE":0x1,
"CE":0x1,"CI":true, "ER":false, "PN":true, "AV":false, "MV":true, "SYN":true,
"R":true}, {"OF":false, "UC":true, "UEU":false, "UER":false, "UEO":false, "DE":0x1,
"CE":0x1,"CI":false, "ER":false, "PN":false, "AV":false, "MV":false, "SYN":false, "R":true}]
```

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

trace_physical_registers_when_host_virtualisation_enabled

When host virtualisation is enabled, trace sysreg accesses to physical register accessed (0=disabled, 1=Trace only ELR/SPSR_EL1 as ELR/SPSR_EL2, 2=Trace all redirected registers as physical registers

Type

int

Default value

0x1

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency
Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type
int

Default value
0x0

3.5.73 ARMCortexX4CT

ARMCortexX4CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-483: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Ports removed:

- l0gptsz

Parameters added:

- mpmm_accumulator_multiplier
- tlbi_stall_enabled

Iris and MTI instances for ARMCortexX4CT

This model has the following Iris instances:

Table 3-484: ARMCortexX4CT Iris instances

InstanceName	ComponentName
ARMCortexX4CT	Cluster_ARM_Cortex-X4
ARMCortexX4CT.AMU	PVBusLogger
ARMCortexX4CT.AMU.mapper	PVBusMapper
ARMCortexX4CT.DAP	PVBusLogger
ARMCortexX4CT.DAP.mapper	PVBusMapper
ARMCortexX4CT.DSU	DSU
ARMCortexX4CT.DSU.PPU_cluster	PPUv1
ARMCortexX4CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexX4CT.DSU.PPU_core0	PPUv1
ARMCortexX4CT.DSU.PPU_core0.busslave	PVBusSlave

InstanceName	ComponentName
ARMCortexX4CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexX4CT.DSU.mpam_busslave	PVBusSlave
ARMCortexX4CT.DSU.shared_cache	PVCache
ARMCortexX4CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexX4CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexX4CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexX4CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexX4CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexX4CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexX4CT.MMAP	PVBusLogger
ARMCortexX4CT.MMAP.mapper	PVBusMapper
ARMCortexX4CT.cpu0	ARM_Cortex-X4
ARMCortexX4CT.cpu0.UTLB	TLB
ARMCortexX4CT.cpu0.dtlb	TlbCadi
ARMCortexX4CT.cpu0.l1dcache	PVCache
ARMCortexX4CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexX4CT.cpu0.l1icache	PVCache
ARMCortexX4CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexX4CT.cpu0.l2cache	PVCache
ARMCortexX4CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexX4CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexX4CT.ext_bus	PVBusLogger
ARMCortexX4CT.ext_bus.mapper	PVBusMapper
ARMCortexX4CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-485: ARMCortexX4CT MTI instances

InstanceName	ComponentName
ARMCortexX4CT	ARMv8Cluster
ARMCortexX4CT.AMU	PVBusLogger
ARMCortexX4CT.AMU.mapper	PVBusMapper
ARMCortexX4CT.DAP	PVBusLogger
ARMCortexX4CT.DAP.mapper	PVBusMapper
ARMCortexX4CT.DSU	DSU
ARMCortexX4CT.DSU.PPU_cluster	PPUv1
ARMCortexX4CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexX4CT.DSU.PPU_core0	PPUv1
ARMCortexX4CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexX4CT.DSU.l3_flusher	AsyncCacheFlushUnit

InstanceName	ComponentName
ARMCortexX4CT.DSU.mpam_busslave	PVBusSlave
ARMCortexX4CT.DSU.shared_cache	PVCache
ARMCortexX4CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexX4CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexX4CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexX4CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexX4CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexX4CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexX4CT.MMAP	PVBusLogger
ARMCortexX4CT.MMAP.mapper	PVBusMapper
ARMCortexX4CT.cpu0	ARM_Cortex-X4
ARMCortexX4CT.cpu0.UTLB	TLB
ARMCortexX4CT.cpu0.l1dcache	PVCache
ARMCortexX4CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexX4CT.cpu0.l1licache	PVCache
ARMCortexX4CT.cpu0.l1licache.upstream[0]	PVBusSlave
ARMCortexX4CT.cpu0.l2cache	PVCache
ARMCortexX4CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexX4CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexX4CT.ext_bus	PVBusLogger
ARMCortexX4CT.ext_bus.mapper	PVBusMapper
ARMCortexX4CT.gic_cpuif_decoder_cluster	GLCv3CPUInterfaceDecoder

ARMCortexX4CT contains the following CADI targets:

- ARM_Cortex-X4
- Cluster_ARM_Cortex-X4
- PVCache
- TlbCadi

About ARMCortexX4CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- DynamIQ r3p0.
- L2 cache is supported at the per-core level only and there is no implementation of Core-Complex with shared L2 cache yet.
- P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.

- Per-core clock.
- `BROADCASTPERSIST` pin.
- Utility bus.
- Support for Interrupt signals from SPE is added as `pmbirq[8]`.

Support for the following features is planned for a future release:

- Transporting architectural metadata tags over TLM2 (OSCI) and AMBA-PV (TLM2 extensions from Arm) bus protocols.
- `BROADCASTCACHEMAINTPOU` pin.
- `COREINSTRRET`, `COREINSTRRUN`, and `nPMBIRQ` signals.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not supported but signals are implemented.



The `cfgsdisable` signal will be removed in a future release.

The following features will not be implemented:

- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Cache stashing capability.
- Embedded Logic Analyzer (ELA).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Limitations

- Some of the SVE instructions do not raise undefined exceptions when SP is used as a source register.

Ports for ARMCortexX4CT

Table 3-486: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[14]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcsn_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[14]	Signal	Master	Timer signals to SOC
CNTHVIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[14]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[14]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[14]	Signal	Master	Timer signals to SOC.

Name	Protocol	Type	Description
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[14]	Signal	Master	Timer signals to SOC.
commirq[14]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[14]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[14]	PChannel	Master	Core PCSM signals
core_powerdown_out[14]	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the core power info
coreerrirq[14]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[14]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[14]	Signal	Slave	Disable cryptography extensions after reset.
cti[14]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[14]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[14]	Signal	Master	No power-down request.
dbgpwrupreq[14]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[14]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPUs that can be used to reset GICCLK domain components
gicv3_redistributor_s[14]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[14]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[14]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[14]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.

Name	Protocol	Type	Description
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[14]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[14]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[14]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[14]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[14]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[14]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[14]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[14]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[14]	Signal	Slave	Virtualised FIQ.
virq[14]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[14]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMCortexX4CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

ecv_support_level

Implement Enhanced Counter Virtualization feature from ARMv8.6. 0, Not supported. 1, fully supported without CNTPOFF. 2, fully supported with CNTPOFF (FEAT_ECV).

Type

int

Default value

0x2

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value

ete.TRCRSRTA_FORCED_EXCEP

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x2

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

has_v8_7_spe_inverted_filtering

Where FEAT_SPEv1p2 is implemented, whether the inverting filtering by events

Type

bool

Default value

0x0

has_v8_7_spe_previous_branch_target

Where FEAT_SPEv1p2 is implemented, whether the optional branch target feature is implemented

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or

miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

log2_trace_buffer_alignment

Log2 of trace buffer alignment constraint for output buffer (0->1B ... 11->2Kib)

Type

int

Default value

0x6

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpam_has_altsp

MPAM Whether MPAMIDR_EL1.HAS_ALTSP bit is set or clear

Type

bool

Default value

0x0

mpamidr_has_force_ns

Whether MPAMIDR_EL1.HAS_FORCE_NS bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

mpamidr_has_sdeflt

Whether MPAMIDR_EL1.HAS_SDEFLT bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

mpamidr_has_tidr

Whether MPAMIDR_EL1.HAS_TIDR bit is set or clear

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_acp

Number of ACP ports

Type

int

Default value

0x0

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x1f

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tcr_txsz_undersize_should_fault

If large VA is not supported, Whether undersized TxSZ value should generate translation fault

Type

bool

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

- tlbi_stall_enabled**

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0
- treat_PAC_as_NOP**

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0
- walk_cache_latency**

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.74 ARMCortexX925CT

ARMCortexX925CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-487: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMCortexX925CT

This model has the following Iris instances:

Table 3-488: ARMCortexX925CT Iris instances

InstanceName	ComponentName
ARMCortexX925CT	Cluster_ARM_Cortex-X925
ARMCortexX925CT.AMU	PVBusLogger
ARMCortexX925CT.AMU.mapper	PVBusMapper
ARMCortexX925CT.DAP	PVBusLogger
ARMCortexX925CT.DAP.mapper	PVBusMapper
ARMCortexX925CT.DSU	DSU
ARMCortexX925CT.DSU.PPU_cluster	PPUv1
ARMCortexX925CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexX925CT.DSU.PPU_core0	PPUv1
ARMCortexX925CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexX925CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexX925CT.DSU.mpam_busslave	PVBusSlave
ARMCortexX925CT.DSU.shared_cache	PVCache
ARMCortexX925CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexX925CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexX925CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexX925CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexX925CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexX925CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexX925CT.MMAP	PVBusLogger
ARMCortexX925CT.MMAP.mapper	PVBusMapper
ARMCortexX925CT.cpu0	ARM_Cortex-X925
ARMCortexX925CT.cpu0.UTLB	TLB
ARMCortexX925CT.cpu0.dtlb	TlbCadi
ARMCortexX925CT.cpu0.l1dcache	PVCache
ARMCortexX925CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexX925CT.cpu0.l1icache	PVCache
ARMCortexX925CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexX925CT.cpu0.l2cache	PVCache
ARMCortexX925CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexX925CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexX925CT.ext_bus	PVBusLogger
ARMCortexX925CT.ext_bus.mapper	PVBusMapper
ARMCortexX925CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-489: ARMCortexX925CT MTI instances

InstanceName	ComponentName
ARMCortexX925CT	ARMv8Cluster
ARMCortexX925CT.AMU	PVBusLogger
ARMCortexX925CT.AMU.mapper	PVBusMapper
ARMCortexX925CT.DAP	PVBusLogger
ARMCortexX925CT.DAP.mapper	PVBusMapper
ARMCortexX925CT.DSU	DSU
ARMCortexX925CT.DSU.PPU_cluster	PPUv1
ARMCortexX925CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMCortexX925CT.DSU.PPU_core0	PPUv1
ARMCortexX925CT.DSU.PPU_core0.busslave	PVBusSlave
ARMCortexX925CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMCortexX925CT.DSU.mpam_busslave	PVBusSlave
ARMCortexX925CT.DSU.shared_cache	PVCache
ARMCortexX925CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMCortexX925CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMCortexX925CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMCortexX925CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMCortexX925CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMCortexX925CT.DSU.utility_slave[0]	PVBusSlave
ARMCortexX925CT.MMAP	PVBusLogger
ARMCortexX925CT.MMAP.mapper	PVBusMapper
ARMCortexX925CT.cpu0	ARM_Cortex-X925
ARMCortexX925CT.cpu0.UTLB	TLB
ARMCortexX925CT.cpu0.l1dcache	PVCache
ARMCortexX925CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMCortexX925CT.cpu0.l1icache	PVCache
ARMCortexX925CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMCortexX925CT.cpu0.l2cache	PVCache
ARMCortexX925CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMCortexX925CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMCortexX925CT.ext_bus	PVBusLogger
ARMCortexX925CT.ext_bus.mapper	PVBusMapper
ARMCortexX925CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMCortexX925CT contains the following CADI targets:

- ARM_Cortex-X925
- Cluster_ARM_Cortex-X925
- PVCache

- TlbCadi

Ports for ARMCortexX925CT

Table 3-490: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[14]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[14]	Signal	Master	Timer signals to SOC
CNTHVIRQ[14]	Signal	Master	Timer signals to SOC.
CNTHVSIQ[14]	Signal	Master	Timer signals to SOC

Name	Protocol	Type	Description
CNTPNSIRQ[14]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[14]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[14]	Signal	Master	Timer signals to SOC.
commirq[14]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[14]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[14]	PChannel	Master	Core PCSM signals
core_powerdown_out[14]	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the core power info
coreerrirq[14]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[14]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[14]	Signal	Slave	Disable cryptography extensions after reset.
cti[14]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[14]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[14]	Signal	Master	No power-down request.
dbgpwrupreq[14]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[14]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPUs that can be used to reset GICCLK domain components
gicv3_redistributor_s[14]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[14]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[14]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[14]	Signal	Master	Interrupt signal from performance monitoring unit.

Name	Protocol	Type	Description
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[14]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[14]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[14]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[14]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[14]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[14]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[14]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[14]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[14]	Signal	Slave	Virtualised FIQ.
virq[14]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[14]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARM CortexX925CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-ways

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

ecv_support_level

Implement Enhanced Counter Virtualization feature from ARMv8.6. 0, Not supported. 1, fully supported without CNTPOFF. 2, fully supported with CNTPOFF (FEAT_ECV).

Type

int

Default value

0x2

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value

ete.TRCRSRTA_FORCED_EXCEP

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x2

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_mt_pmu_disable_feature

Implements Multi-threading PMU disable extension from ARMv8.6 (FEAT_MTPMU). 0: FEAT_MTPMU is disabled, 1: FEAT_MTPMU is enabled if ARMv8.6 is implemented, 2: FEAT_MTPMU is cherry-picked, 0xF: The feature is disabled and is represented by value 0xF in ID_AA64DFR0_EL1.MTPMU.

Type

int

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

has_v8_7_spe_inverted_filtering

Where FEAT_SPEv1p2 is implemented, whether the inverting filtering by events

Type

bool

Default value

0x0

has_v8_7_spe_previous_branch_target

Where FEAT_SPEv1p2 is implemented, whether the optional branch target feature is implemented

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x80000

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

log2_trace_buffer_alignment

Log2 of trace buffer alignment constraint for output buffer (0->1B ... 11->2Kib)

Type

int

Default value

0x6

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpam_has_altsp

MPAM Whether MPAMIDR_EL1.HAS_ALTSP bit is set or clear

Type

bool

Default value

0x0

mpamidr_has_force_ns

Whether MPAMIDR_EL1.HAS_FORCE_NS bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

mpamidr_has_sdeflt

Whether MPAMIDR_EL1.HAS_SDEFLT bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

mpamidr_has_tidr

Whether MPAMIDR_EL1.HAS_TIDR bit is set or clear

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_acp

Number of ACP ports

Type

int

Default value

0x0

num_nodes

Number of transport nodes. Zero implies direct-connect configuration

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x1f

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tcr_txsz_undersize_should_fault

If large VA is not supported, Whether undersized TxSZ value should generate translation fault

Type

bool

Default value

0x0

tlb_latency
TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type
int

Default value
0x0

tlbi_stall_enabled
If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type
bool

Default value
0x0

treat_PAC_as_NOP
Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type
bool

Default value
0x0

walk_cache_latency
Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type
int

Default value
0x0

3.5.75 ARMNeoverseE1CT

ARMNeoverseE1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-491: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `tlbi_stall_enabled`

Iris and MTI instances for ARMNeoverseE1CT

This model has the following Iris instances:

Table 3-492: ARMNeoverseE1CT Iris instances

InstanceName	ComponentName
ARMNeoverseE1CT	Cluster_ARM_Neoverse-E1
ARMNeoverseE1CT.AMU	PVBusLogger
ARMNeoverseE1CT.AMU.mapper	PVBusMapper
ARMNeoverseE1CT.DAP	PVBusLogger
ARMNeoverseE1CT.DAP.mapper	PVBusMapper
ARMNeoverseE1CT.DSU	DSU
ARMNeoverseE1CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMNeoverseE1CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseE1CT.DSU.shared_cache	PVCache
ARMNeoverseE1CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseE1CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseE1CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseE1CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseE1CT.MMAP	PVBusLogger
ARMNeoverseE1CT.MMAP.mapper	PVBusMapper
ARMNeoverseE1CT.cpu0.dtlb	TlbCadi
ARMNeoverseE1CT.cpu0.thread0	ARM_Neoverse-E1
ARMNeoverseE1CT.cpu0.thread0.UTLB	TLB
ARMNeoverseE1CT.cpu0.thread0.l1dcache	PVCache
ARMNeoverseE1CT.cpu0.thread0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseE1CT.cpu0.thread0.l1lcache	PVCache
ARMNeoverseE1CT.cpu0.thread0.l1lcache.upstream[0]	PVBusSlave
ARMNeoverseE1CT.cpu0.thread0.l2cache	PVCache
ARMNeoverseE1CT.cpu0.thread0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseE1CT.cpu0.thread0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseE1CT.cpu0.thread1	ARM_Neoverse-E1
ARMNeoverseE1CT.cpu0.thread1.UTLB	TLB
ARMNeoverseE1CT.ext_bus	PVBusLogger
ARMNeoverseE1CT.ext_bus.mapper	PVBusMapper
ARMNeoverseE1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-493: ARMNeoverseE1CT MTI instances

InstanceName	ComponentName
ARMNeoverseE1CT	ARMv8Cluster
ARMNeoverseE1CT.AMU	PVBusLogger
ARMNeoverseE1CT.AMU.mapper	PVBusMapper
ARMNeoverseE1CT.DAP	PVBusLogger
ARMNeoverseE1CT.DAP.mapper	PVBusMapper
ARMNeoverseE1CT.DSU	DSU
ARMNeoverseE1CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMNeoverseE1CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseE1CT.DSU.shared_cache	PVCache
ARMNeoverseE1CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseE1CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseE1CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseE1CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseE1CT.MMAP	PVBusLogger
ARMNeoverseE1CT.MMAP.mapper	PVBusMapper
ARMNeoverseE1CT.cpu0.thread0	ARM_Neoverse-E1
ARMNeoverseE1CT.cpu0.thread0.UTLB	TLB
ARMNeoverseE1CT.cpu0.thread0.l1dcache	PVCache
ARMNeoverseE1CT.cpu0.thread0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseE1CT.cpu0.thread0.l1icache	PVCache
ARMNeoverseE1CT.cpu0.thread0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseE1CT.cpu0.thread0.l2cache	PVCache
ARMNeoverseE1CT.cpu0.thread0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseE1CT.cpu0.thread0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseE1CT.cpu0.thread1	ARM_Neoverse-E1
ARMNeoverseE1CT.cpu0.thread1.UTLB	TLB
ARMNeoverseE1CT.ext_bus	PVBusLogger
ARMNeoverseE1CT.ext_bus.mapper	PVBusMapper
ARMNeoverseE1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMNeoverseE1CT contains the following CADI targets:

- ARM_Neoverse-E1
- Cluster_ARM_Neoverse-E1
- PVCache
- TlbCadi

About ARMNeoverseE1CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, or nPMBIRQ signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals DBGPWRDUP and DBGIRSTREQ are not implemented, but DBGPWRUPREQ and DBGNOFWRDWN are implemented.
- Cache stashing capability.
- Each thread has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.

This model has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following models also exist, with a fixed number of cores per cluster:

- ARMNeoverseE1x1CT.
- ARMNeoverseE1x2CT.
- ARMNeoverseE1x3CT.
- ARMNeoverseE1x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (crypto.dll or crypto.so) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMNeoverseE1CT

Table 3-494: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port
AENDMP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTARTMP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[16]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[16]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[16]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[16]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[16]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[16]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[16]	Signal	Master	Timer signals to SOC.
commirq[16]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[8]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cp15sdisable[16]	Signal	Slave	This signal disables write access to some system control processor registers.
cpuporeset[8]	Signal	Slave	Power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[16]	Signal	Slave	Disable cryptography extensions after reset.
cti[16]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.

Name	Protocol	Type	Description
ctidbgirq[16]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[16]	Signal	Master	These signals relate to core power down.
dbgpwrupreq[16]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[16]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[16]	GICv3Comms	Slave	GICv3 AXI-stream port per thread.
irq[16]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[8]	PChannel	Slave	PChannels for cores
pmuirq[16]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[16]	Signal	Slave	Per core RAM Error Interrupt.
reset[8]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[16]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[16]	Signal	Slave	Per core virtual System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynaMIQ.
ticks[16]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[16]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.

Name	Protocol	Type	Description
vfiq[16]	Signal	Slave	Virtualised FIQ.
vinithi[16]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[16]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[16]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMNeoverseE1CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cluster_patch_level

Cosmetic change to patch number in CLUSTERIDR. Corresponds to the Y in rXpY.

Type

int

Default value

0x0

cluster_revision_number

Cosmetic change to revision number in CLUSTERIDR, Corresponds to the X in rXpY.

Type

int

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.enable_single_thread_at_reset

Enable single thread after reset and keep other thread in reset

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x40000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.threadY.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.threadY.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.threadY.MPIDR-override

Override MPIDR value. A nonzero value will override the MT,cluster,cpu,thread ID bits in MPIDR.

Type

int

Default value

0x0

cpuX.threadY.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.threadY.VINITI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_dot_product

Possible values of this parameter are: - 1, feature not implemented. - 2, feature is implemented.

Type

int

Default value

0x2

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x400000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.76 ARMNeoverseN1CT

ARMNeoverseN1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-495: IP revisions support

Revision	Quality level
r4p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- tlbi_stall_enabled

Iris and MTI instances for ARMNeoverseN1CT

This model has the following Iris instances:

Table 3-496: ARMNeoverseN1CT Iris instances

InstanceName	ComponentName
ARMNeoverseN1CT	Cluster_ARM_Neoverse-N1

InstanceName	ComponentName
ARMNeoverseN1CT.AMU	PVBusLogger
ARMNeoverseN1CT.AMU.mapper	PVBusMapper
ARMNeoverseN1CT.DAP	PVBusLogger
ARMNeoverseN1CT.DAP.mapper	PVBusMapper
ARMNeoverseN1CT.DSU	DSU
ARMNeoverseN1CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMNeoverseN1CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseN1CT.DSU.shared_cache	PVCache
ARMNeoverseN1CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseN1CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseN1CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseN1CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseN1CT.MMAP	PVBusLogger
ARMNeoverseN1CT.MMAP.mapper	PVBusMapper
ARMNeoverseN1CT.cpu0	ARM_Neoverse-N1
ARMNeoverseN1CT.cpu0.UTLB	TLB
ARMNeoverseN1CT.cpu0.dtlb	TlbCadi
ARMNeoverseN1CT.cpu0.l1dcache	PVCache
ARMNeoverseN1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseN1CT.cpu0.l1icache	PVCache
ARMNeoverseN1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseN1CT.cpu0.l2cache	PVCache
ARMNeoverseN1CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseN1CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseN1CT.ext_bus	PVBusLogger
ARMNeoverseN1CT.ext_bus.mapper	PVBusMapper
ARMNeoverseN1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-497: ARMNeoverseN1CT MTI instances

InstanceName	ComponentName
ARMNeoverseN1CT	ARMv8Cluster
ARMNeoverseN1CT.AMU	PVBusLogger
ARMNeoverseN1CT.AMU.mapper	PVBusMapper
ARMNeoverseN1CT.DAP	PVBusLogger
ARMNeoverseN1CT.DAP.mapper	PVBusMapper
ARMNeoverseN1CT.DSU	DSU
ARMNeoverseN1CT.DSU.l3_flusher	AsyncCacheFlushUnit
ARMNeoverseN1CT.DSU.mpam_busslave	PVBusSlave

InstanceName	ComponentName
ARMNeoverseN1CT.DSU.shared_cache	PVCache
ARMNeoverseN1CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseN1CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseN1CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseN1CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseN1CT.MMAP	PVBusLogger
ARMNeoverseN1CT.MMAP.mapper	PVBusMapper
ARMNeoverseN1CT.cpu0	ARM_Neoverse-N1
ARMNeoverseN1CT.cpu0.UTLB	TLB
ARMNeoverseN1CT.cpu0.l1dcache	PVCache
ARMNeoverseN1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseN1CT.cpu0.l1icache	PVCache
ARMNeoverseN1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseN1CT.cpu0.l2cache	PVCache
ARMNeoverseN1CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseN1CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseN1CT.ext_bus	PVBusLogger
ARMNeoverseN1CT.ext_bus.mapper	PVBusMapper
ARMNeoverseN1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMNeoverseN1CT contains the following CADI targets:

- ARM_Neoverse-N1
- Cluster_ARM_Neoverse-N1
- PVCache
- TlbCadi

About ARMNeoverseN1CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

The model does not support the following features:

- BROADCASTCACHEMAINTPOU pin.

- `COREINSTRET`, `COREINSTRUN`, or `nPMBIRQ` signals.
- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.
- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Latency configuration.
- Snoop filtering.
- Debug power management signals `DBGPWRDUP` and `DBGIRSTREQ` are not implemented, but `DBGPWRUPREQ` and `DBGNOPWRDWN` are implemented.
- Cache stashing capability.

This model has a variable number of cores per cluster, specified using the `NUM_CORES` parameter. The following models also exist, with a fixed number of cores per cluster:

- ARMNeoverseN1x1CT.
- ARMNeoverseN1x2CT.
- ARMNeoverseN1x3CT.
- ARMNeoverseN1x4CT.

The per-core parameters are preceded by `cpun.`, where *n* identifies the core (0-3).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMNeoverseN1CT

Table 3-498: Ports

Name	Protocol	Type	Description
<code>acp_s</code>	PVBus	Slave	AXI ACP slave port.
<code>AENDMP</code>	<code>Value_64</code>	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTARTMP</code>	<code>Value_64</code>	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[4]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.

Name	Protocol	Type	Description
cfgte[4]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[4]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[4]	Signal	Master	Timer signals to SOC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[4]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cpuporeset[4]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[4]	PChannel	Slave	PChannels for cores
pmbirq[4]	Signal	Master	Interrupt signal from the statistical profiling unit.

Name	Protocol	Type	Description
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[4]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMNeoverseN1CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x0

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x100000

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.77 ARMNeoverseN2CT

ARMNeoverseN2CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-499: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `mpmm_accumulator_multiplier`
- `tlbi_stall_enabled`

Iris and MTI instances for ARMNeoverseN2CT

This model has the following Iris instances:

Table 3-500: ARMNeoverseN2CT Iris instances

InstanceName	ComponentName
ARMNeoverseN2CT	Cluster_ARM_Neoverse-N2
ARMNeoverseN2CT.AMU	PVBusLogger
ARMNeoverseN2CT.AMU.mapper	PVBusMapper
ARMNeoverseN2CT.DAP	PVBusLogger
ARMNeoverseN2CT.DAP.mapper	PVBusMapper
ARMNeoverseN2CT.DSU	DSU
ARMNeoverseN2CT.DSU.PPU_cluster	PPUv1
ARMNeoverseN2CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMNeoverseN2CT.DSU.PPU_core0	PPUv1
ARMNeoverseN2CT.DSU.PPU_core0.busslave	PVBusSlave
ARMNeoverseN2CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseN2CT.DSU.shared_cache	PVCache
ARMNeoverseN2CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseN2CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseN2CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseN2CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseN2CT.DSU.utility_slave[0]	PVBusSlave

InstanceName	ComponentName
ARMNeoverseN2CT.MMAP	PVBusLogger
ARMNeoverseN2CT.MMAP.mapper	PVBusMapper
ARMNeoverseN2CT.cpu0	ARM_Neoverse-N2
ARMNeoverseN2CT.cpu0.UTLB	TLB
ARMNeoverseN2CT.cpu0.dtlb	TlbCadi
ARMNeoverseN2CT.cpu0.l1dcache	PVCache
ARMNeoverseN2CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseN2CT.cpu0.l1icache	PVCache
ARMNeoverseN2CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseN2CT.cpu0.l2cache	PVCache
ARMNeoverseN2CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseN2CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseN2CT.ext_bus	PVBusLogger
ARMNeoverseN2CT.ext_bus.mapper	PVBusMapper
ARMNeoverseN2CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-501: ARMNeoverseN2CT MTI instances

InstanceName	ComponentName
ARMNeoverseN2CT	ARMv8Cluster
ARMNeoverseN2CT.AMU	PVBusLogger
ARMNeoverseN2CT.AMU.mapper	PVBusMapper
ARMNeoverseN2CT.DAP	PVBusLogger
ARMNeoverseN2CT.DAP.mapper	PVBusMapper
ARMNeoverseN2CT.DSU	DSU
ARMNeoverseN2CT.DSU.PPU_cluster	PPUv1
ARMNeoverseN2CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMNeoverseN2CT.DSU.PPU_core0	PPUv1
ARMNeoverseN2CT.DSU.PPU_core0.busslave	PVBusSlave
ARMNeoverseN2CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseN2CT.DSU.shared_cache	PVCache
ARMNeoverseN2CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseN2CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseN2CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseN2CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseN2CT.DSU.utility_slave[0]	PVBusSlave
ARMNeoverseN2CT.MMAP	PVBusLogger
ARMNeoverseN2CT.MMAP.mapper	PVBusMapper
ARMNeoverseN2CT.cpu0	ARM_Neoverse-N2

InstanceName	ComponentName
ARMNeoverseN2CT.cpu0.UTLB	TLB
ARMNeoverseN2CT.cpu0.l1dcache	PVCache
ARMNeoverseN2CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseN2CT.cpu0.l1icache	PVCache
ARMNeoverseN2CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseN2CT.cpu0.l2cache	PVCache
ARMNeoverseN2CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseN2CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseN2CT.ext_bus	PVBusLogger
ARMNeoverseN2CT.ext_bus.mapper	PVBusMapper
ARMNeoverseN2CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMNeoverseN2CT contains the following CADI targets:

- ARM_Neoverse-N2
- Cluster_ARM_Neoverse-N2
- PVCache
- TlbCadi

About ARMNeoverseN2CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

Support for the following features is planned for a future release:

- DynamIQ Shared Unit-110 (DSU-110) system registers.
- BROADCASTCACHEMAINTPOU pin
- COREINSTRRET, COREINSTRRUN, and nPMBIRQ signals
- DSU-110 cluster. The implementation relies on DynamIQ only.
- TRBE.
- Transporting architectural metadata tags over TLM2 (OSCI) and AMBA-PV (the Arm TLM2 extensions) bus protocols.

The following features will not be implemented:

- 256-bit wide output transactions.
- Error correction/detection features.
- Self-test features (MBIST).
- Snoop filtering.
- Armv9 trace extensions.

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMNeoverseN2CT

Table 3-502: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[1]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal

Name	Protocol	Type	Description
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[1]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[1]	Signal	Master	Timer signals to SOC
CNTHVIRQ[1]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[1]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[1]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[1]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[1]	Signal	Master	Timer signals to SOC.
commirq[1]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[1]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[1]	PChannel	Master	Core PCSM signals
core_powerdown_out[1]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[1]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[1]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[1]	Signal	Slave	Disable cryptography extensions after reset.
cti[1]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[1]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[1]	Signal	Master	No power-down request.
dbgpwrupreq[1]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[1]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.

Name	Protocol	Type	Description
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[1]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[1]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[1]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[1]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[1]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[1]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[1]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[1]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[1]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[1]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[1]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[1]	Signal	Slave	Virtualised FIQ.
virq[1]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).

Name	Protocol	Type	Description
vsei[1]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMNeoverseN2CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x100000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`dcache-snoop_data_transfer_latency`

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`dcache-state_modelled`

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

`dcache-write_access_latency`

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size `dcache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `dcache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`dcache-write_latency`

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. `dcache-write_access_latency` must be set to 0 for per-byte latencies to be applied. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x20

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value

ete.TRCRSRTA_FORCED_EXCEP

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_read_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x0

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_external_rndr

Implement external random number generator module. When enabling this with has_rndr enabled, the external random number generator will be used instead of internal random number generator

Type

int

Default value

0x1

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_rndr

Implement random number instructions to read from RNDR and RNDRSS random number registers from ARMv8.5 (FEAT_RNG).

Type

int

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied.

This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

`icache-state_modelled`

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

`l3cache-hit_latency`

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-maintenance_latency`

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-miss_latency`

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-read_access_latency`

L3 Cache timing annotation latency for read accesses given in ticks per access (of size `l3cache-read_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `l3cache-read_latency` is set. This is in addition to the hit

or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-read_latency`

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l3cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-snoop_data_transfer_latency`

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-snoop_issue_latency`

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`l3cache-write_access_latency`

L3 Cache timing annotation latency for write accesses given in ticks per access (of size `l3cache-write_bus_width_in_bytes`). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if `l3cache-write_latency` is set. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).

Type

int

Default value

0x2

mpam_max_partid

MPAM Maximum PARTID Supported

Type

int

Default value

0x1fff

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

rndr_rndrrs_seed

Initial seed for random engine used in RNDR register

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.78 ARMNeoverseN3CT

ARMNeoverseN3CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-503: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMNeoverseN3CT

This model has the following Iris instances:

Table 3-504: ARMNeoverseN3CT Iris instances

InstanceName	ComponentName
ARMNeoverseN3CT	Cluster_ARM_Neoverse-N3
ARMNeoverseN3CT.AMU	PVBusLogger
ARMNeoverseN3CT.AMU.mapper	PVBusMapper
ARMNeoverseN3CT.DAP	PVBusLogger

InstanceName	ComponentName
ARMNeoverseN3CT.DAP.mapper	PVBusMapper
ARMNeoverseN3CT.DSU	DSU
ARMNeoverseN3CT.DSU.PPU_cluster	PPUv1
ARMNeoverseN3CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMNeoverseN3CT.DSU.PPU_core0	PPUv1
ARMNeoverseN3CT.DSU.PPU_core0.busslave	PVBusSlave
ARMNeoverseN3CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseN3CT.DSU.shared_cache	PVCache
ARMNeoverseN3CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseN3CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseN3CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseN3CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseN3CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMNeoverseN3CT.DSU.utility_slave[0]	PVBusSlave
ARMNeoverseN3CT.MMAP	PVBusLogger
ARMNeoverseN3CT.MMAP.mapper	PVBusMapper
ARMNeoverseN3CT.cpu0	ARM_Neoverse-N3
ARMNeoverseN3CT.cpu0.UTLB	TLB
ARMNeoverseN3CT.cpu0.dtlb	TlbCadi
ARMNeoverseN3CT.cpu0.l1dcache	PVCache
ARMNeoverseN3CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseN3CT.cpu0.l1icache	PVCache
ARMNeoverseN3CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseN3CT.cpu0.l2cache	PVCache
ARMNeoverseN3CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseN3CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseN3CT.ext_bus	PVBusLogger
ARMNeoverseN3CT.ext_bus.mapper	PVBusMapper
ARMNeoverseN3CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-505: ARMNeoverseN3CT MTI instances

InstanceName	ComponentName
ARMNeoverseN3CT	ARMv8Cluster
ARMNeoverseN3CT.AMU	PVBusLogger
ARMNeoverseN3CT.AMU.mapper	PVBusMapper
ARMNeoverseN3CT.DAP	PVBusLogger
ARMNeoverseN3CT.DAP.mapper	PVBusMapper
ARMNeoverseN3CT.DSU	DSU

InstanceName	ComponentName
ARMNeoverseN3CT.DSU.PPU_cluster	PPUv1
ARMNeoverseN3CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMNeoverseN3CT.DSU.PPU_core0	PPUv1
ARMNeoverseN3CT.DSU.PPU_core0.busslave	PVBusSlave
ARMNeoverseN3CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseN3CT.DSU.shared_cache	PVCache
ARMNeoverseN3CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseN3CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseN3CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseN3CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseN3CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMNeoverseN3CT.DSU.utility_slave[0]	PVBusSlave
ARMNeoverseN3CT.MMAP	PVBusLogger
ARMNeoverseN3CT.MMAP.mapper	PVBusMapper
ARMNeoverseN3CT.cpu0	ARM_Neoverse-N3
ARMNeoverseN3CT.cpu0.UTLB	TLB
ARMNeoverseN3CT.cpu0.l1dcache	PVCache
ARMNeoverseN3CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseN3CT.cpu0.l1icache	PVCache
ARMNeoverseN3CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseN3CT.cpu0.l2cache	PVCache
ARMNeoverseN3CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseN3CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseN3CT.ext_bus	PVBusLogger
ARMNeoverseN3CT.ext_bus.mapper	PVBusMapper
ARMNeoverseN3CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMNeoverseN3CT contains the following CADI targets:

- ARM_Neoverse-N3
- Cluster_ARM_Neoverse-N3
- PVCache
- TlbCadi

Ports for ARMNeoverseN3CT

Table 3-506: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).

Name	Protocol	Type	Description
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[1]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[1]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[1]	Signal	Master	Timer signals to SOC
CNTHVIRQ[1]	Signal	Master	Timer signals to SOC.
CNTHVSIQ[1]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[1]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[1]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[1]	Signal	Master	Timer signals to SOC.
commirq[1]	Signal	Master	Interrupt signal from debug communications channel.

Name	Protocol	Type	Description
core_clk_in[1]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcsmpchannel[1]	PChannel	Master	Core PCSM signals
core_powerdown_out[1]	Signal	Master	DEPRECATED - An output from PPUs that informs thermal controller of the core power info
coreerrirq[1]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[1]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[1]	Signal	Slave	Disable cryptography extensions after reset.
cti[1]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[1]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[1]	Signal	Master	No power-down request.
dbgpwrupreq[1]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[1]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPUs that can be used to reset GICCLK domain components
gicv3_redistributor_s[1]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[1]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[1]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[1]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[1]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[1]	Signal	Slave	PPU core wake request signal.

Name	Protocol	Type	Description
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[1]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[1]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[1]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[1]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[1]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[1]	Signal	Slave	Virtualised FIQ.
virq[1]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[1]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMNeoverseN3CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x4

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x100000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.ETE_REVISION

ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3

Type

int

Default value

0x1

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x1

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

ete.TSMARK

Whether timestamp markers are supported

Type

bool

Default value

0x1

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance

of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_rndr_trap

Implement trapping for RNDR and RNDRSS random number registers from ARMv8.8. (FEAT_RNG_TRAP)

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.8 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.79 ARMNeoverseV1CT

ARMNeoverseV1CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-507: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `tlbi_stall_enabled`

Parameters removed:

- `MIDR`

Iris and MTI instances for ARMNeoverseV1CT

This model has the following Iris instances:

Table 3-508: ARMNeoverseV1CT Iris instances

InstanceName	ComponentName
ARMNeoverseV1CT	Cluster_ARM_Neoverse-V1
ARMNeoverseV1CT.AMU	PVBusLogger
ARMNeoverseV1CT.AMU.mapper	PVBusMapper
ARMNeoverseV1CT.DAP	PVBusLogger
ARMNeoverseV1CT.DAP.mapper	PVBusMapper
ARMNeoverseV1CT.DSU	DSU
ARMNeoverseV1CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseV1CT.DSU.shared_cache	PVCache
ARMNeoverseV1CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseV1CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseV1CT.DSU.shared_cache.upstream[2]	PVBusSlave

InstanceName	ComponentName
ARMNeoverseV1CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseV1CT.MMAP	PVBusLogger
ARMNeoverseV1CT.MMAP.mapper	PVBusMapper
ARMNeoverseV1CT.cpu0	ARM_Neoverse-V1
ARMNeoverseV1CT.cpu0.UTLB	TLB
ARMNeoverseV1CT.cpu0.dtlb	TlbCadi
ARMNeoverseV1CT.cpu0.l1dcache	PVCache
ARMNeoverseV1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseV1CT.cpu0.l1icache	PVCache
ARMNeoverseV1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseV1CT.cpu0.l2cache	PVCache
ARMNeoverseV1CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseV1CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseV1CT.ext_bus	PVBusLogger
ARMNeoverseV1CT.ext_bus.mapper	PVBusMapper
ARMNeoverseV1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-509: ARMNeoverseV1CT MTI instances

InstanceName	ComponentName
ARMNeoverseV1CT	ARMv8Cluster
ARMNeoverseV1CT.AMU	PVBusLogger
ARMNeoverseV1CT.AMU.mapper	PVBusMapper
ARMNeoverseV1CT.DAP	PVBusLogger
ARMNeoverseV1CT.DAP.mapper	PVBusMapper
ARMNeoverseV1CT.DSU	DSU
ARMNeoverseV1CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseV1CT.DSU.shared_cache	PVCache
ARMNeoverseV1CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseV1CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseV1CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseV1CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseV1CT.MMAP	PVBusLogger
ARMNeoverseV1CT.MMAP.mapper	PVBusMapper
ARMNeoverseV1CT.cpu0	ARM_Neoverse-V1
ARMNeoverseV1CT.cpu0.UTLB	TLB
ARMNeoverseV1CT.cpu0.l1dcache	PVCache
ARMNeoverseV1CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseV1CT.cpu0.l1icache	PVCache

InstanceName	ComponentName
ARMNeoverseV1CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseV1CT.cpu0.l2cache	PVCache
ARMNeoverseV1CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseV1CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseV1CT.ext_bus	PVBusLogger
ARMNeoverseV1CT.ext_bus.mapper	PVBusMapper
ARMNeoverseV1CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMNeoverseV1CT contains the following CADI targets:

- ARM_Neoverse-V1
- Cluster_ARM_Neoverse-V1
- PVCache
- TlbCadi

About ARMNeoverseV1CT

The model supports the following features:

- DynamIQ™ (DSU) system registers.
- Per-core L2 cache.
- A P-Channel for the cluster and for each core.
- Optional peripheral port.
- L3 cache partition.
- Per-core clock.
- BROADCASTPERSIST pin.

Support for the following features is planned for a future release:

- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, and nPMBIRQ signals.
- A common cache that is shared by all threads of the core. Currently, each thread has its own L1 cache and L2 cache.
- Per-thread parameters, although signals are implemented.

The following features will not be implemented:

- 256-bit wide output transactions.
- ECC and parity schemes are hardware-specific so are not supported.
- Self-test features (MBIST).
- Snoop filtering.
- features that are irrelevant to the programmers' view simulation, for example:

- Automatic CPU retention mode.
- Level-3 Cache RAM retention.
- Latency configuration.
- Cache stashing capability.

The following models also exist, with a fixed number of cores per cluster:

- ARMNeoverseV1x1CT.
- ARMNeoverseV1x2CT.
- ARMNeoverseV1x4CT.

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

Ports for ARMNeoverseV1CT

Table 3-510: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
AENDMP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTARTMP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[4]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
cfgte[4]	Signal	Slave	This signal provides default exception handling state.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[4]	Signal	Master	Timer signals to SOC.
CNTHVIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPNSIRQ[4]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[4]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.

Name	Protocol	Type	Description
CNTVIRQ[4]	Signal	Master	Timer signals to SOC.
commirq[4]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[4]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
cpuporeset[4]	Signal	Slave	CPU power on reset. Initializes all the processor logic, including debug logic.
cryptodisable[4]	Signal	Slave	Disable cryptography extensions after reset.
cti[4]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[4]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[4]	Signal	Master	No power-down request.
dbgpwrupreq[4]	Signal	Master	Debug power up request.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[4]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicv3_redistributor_s[4]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[4]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
niden	Signal	Slave	External debug interface.
pchannel_cluster	PChannel	Slave	PChannel for cluster.
pchannel_core[4]	PChannel	Slave	PChannels for cores
pmbirq[4]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[4]	Signal	Master	Interrupt signal from performance monitoring unit.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[4]	Signal	Slave	Per core RAM Error Interrupt.
reset[4]	Signal	Slave	Raising this signal will put the core into reset mode.
rvbaraddr[4]	Value_64	Slave	Reset vector base address.

Name	Protocol	Type	Description
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[4]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
spniden	Signal	Slave	External debug interface.
sporeset	Signal	Slave	A single cluster-wide power on reset signal for all resettable registers in DynamIQ.
ticks[4]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
vcpumntirq[4]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[4]	Signal	Slave	Virtualised FIQ.
vinithi[4]	Signal	Slave	This signal controls of the location of the exception vectors at reset.
virq[4]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[4]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMNeoverseV1CT

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x1

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

cpi_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.VINITHI

Reset value of SCTLR.V.

Type

bool

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x80000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value

cpuX.semihosting-cwd

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

enhanced_pac2_level

Implements Enhanced PAC2 from ARMv8.6 (FEAT_PAuth2), and PAC enhancements from ARMv9.5 (FEAT_PAuth_LR). options 0-3 of this feature are mandatory for ARMv8.6 but can be cherry-picked to a ARMv8.3(or greater) implementation. 0: No EnhancedPAC2, 1: EnhancedPAC2 Only (FEAT_PAuth2), 2: EnhancedPAC2 with FPAC (FEAT_FPAC), 3: EnhancedPAC2 with FPACCombined (FEAT_FPACCOMBINE), 4: EnhancedPAC2 with LR signing (FEAT_PAuth_LR).

Type

int

Default value

0x1

ext_abort_device_read_is_sync

Synchronous reporting of device-nGnRE read external aborts

Type

bool

Default value

0x0

ext_abort_device_write_is_sync

Synchronous reporting of device-nGnRE write external aborts

Type

bool

Default value

0x0

ext_abort_so_read_is_sync

Synchronous reporting of device-nGnRnE read external aborts

Type

bool

Default value

0x0

ext_abort_so_write_is_sync

Synchronous reporting of device-nGnRnE write external aborts

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_acp

If true, Accelerator Coherency Port is configured

Type

bool

Default value

0x0

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

has_external_rndr

Implement external random number generator module. When enabling this with has_rndr enabled, the external random number generator will be used instead of internal random number generator

Type

int

Default value

0x1

has_peripheral_port

If true, additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_rndr

Implement random number instructions to read from RNDR and RNDRSS random number registers from ARMv8.5 (FEAT_RNG).

Type

int

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. icache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

l3cache-hit_latency

L3 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-maintenance_latency

L3 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-miss_latency

L3 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_access_latency

L3 Cache timing annotation latency for read accesses given in ticks per access (of size l3cache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-read_latency

L3 Cache timing annotation latency for read accesses given in ticks per byte accessed. l3cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-size

L3 Cache size in bytes.

Type

int

Default value

0x0

l3cache-snoop_data_transfer_latency

L3 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-snoop_issue_latency

L3 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_access_latency

L3 Cache timing annotation latency for write accesses given in ticks per access (of size l3cache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l3cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

l3cache-write_latency

L3 Cache timing annotation latency for write accesses given in ticks per byte accessed. l3cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

pchannel_treat_simreset_as_poreset

Register core as ON state to cluster with simulation reset.

Type

bool

Default value

0x0

periph_address_end

End address for peripheral port address range exclusive(corresponds to AENDMP input signal).

Type

int

Default value

0x0

periph_address_start

Start address for peripheral port address range inclusive(corresponds to ASTARTMP input signal).

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat-dcache-cmos-to-pou-as-nop

Whether dcache invalidation to the point of unification is required for instruction to data coherence. 0 - Invalidate ops required, 1 - Invalidate ops not required and cannot generate faults, 2 - Invalidate ops not required but can generate faults

Type

int

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.80 ARMNeoverseV2CT

ARMNeoverseV2CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-511: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `mpmm_accumulator_multiplier`
- `tlbi_stall_enabled`

Iris and MTI instances for ARMNeoverseV2CT

This model has the following Iris instances:

Table 3-512: ARMNeoverseV2CT Iris instances

InstanceName	ComponentName
ARMNeoverseV2CT	Cluster_ARM_Neoverse-V2
ARMNeoverseV2CT.AMU	PVBusLogger
ARMNeoverseV2CT.AMU.mapper	PVBusMapper
ARMNeoverseV2CT.DAP	PVBusLogger
ARMNeoverseV2CT.DAP.mapper	PVBusMapper
ARMNeoverseV2CT.DSU	DSU
ARMNeoverseV2CT.DSU.PPU_cluster	PPUv1
ARMNeoverseV2CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMNeoverseV2CT.DSU.PPU_core0	PPUv1
ARMNeoverseV2CT.DSU.PPU_core0.busslave	PVBusSlave
ARMNeoverseV2CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseV2CT.DSU.shared_cache	PVCache
ARMNeoverseV2CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseV2CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseV2CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseV2CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseV2CT.DSU.utility_slave[0]	PVBusSlave
ARMNeoverseV2CT.MMAP	PVBusLogger
ARMNeoverseV2CT.MMAP.mapper	PVBusMapper
ARMNeoverseV2CT.cpu0	ARM_Neoverse-V2
ARMNeoverseV2CT.cpu0.UTLB	TLB
ARMNeoverseV2CT.cpu0.dtlb	TlbCadi
ARMNeoverseV2CT.cpu0.l1dcache	PVCache
ARMNeoverseV2CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseV2CT.cpu0.l1icache	PVCache

InstanceName	ComponentName
ARMNeoverseV2CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseV2CT.cpu0.l2cache	PVCache
ARMNeoverseV2CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseV2CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseV2CT.ext_bus	PVBusLogger
ARMNeoverseV2CT.ext_bus.mapper	PVBusMapper
ARMNeoverseV2CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-513: ARMNeoverseV2CT MTI instances

InstanceName	ComponentName
ARMNeoverseV2CT	ARMv8Cluster
ARMNeoverseV2CT.AMU	PVBusLogger
ARMNeoverseV2CT.AMU.mapper	PVBusMapper
ARMNeoverseV2CT.DAP	PVBusLogger
ARMNeoverseV2CT.DAP.mapper	PVBusMapper
ARMNeoverseV2CT.DSU	DSU
ARMNeoverseV2CT.DSU.PPU_cluster	PPUv1
ARMNeoverseV2CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMNeoverseV2CT.DSU.PPU_core0	PPUv1
ARMNeoverseV2CT.DSU.PPU_core0.busslave	PVBusSlave
ARMNeoverseV2CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseV2CT.DSU.shared_cache	PVCache
ARMNeoverseV2CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseV2CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseV2CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseV2CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseV2CT.DSU.utility_slave[0]	PVBusSlave
ARMNeoverseV2CT.MMAP	PVBusLogger
ARMNeoverseV2CT.MMAP.mapper	PVBusMapper
ARMNeoverseV2CT.cpu0	ARM_Neoverse-V2
ARMNeoverseV2CT.cpu0.UTLB	TLB
ARMNeoverseV2CT.cpu0.l1dcache	PVCache
ARMNeoverseV2CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseV2CT.cpu0.l1icache	PVCache
ARMNeoverseV2CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseV2CT.cpu0.l2cache	PVCache
ARMNeoverseV2CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseV2CT.cpu0.l2cache.upstream[1]	PVBusSlave

InstanceName	ComponentName
ARMNeoverseV2CT.ext_bus	PVBusLogger
ARMNeoverseV2CT.ext_bus.mapper	PVBusMapper
ARMNeoverseV2CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMNeoverseV2CT contains the following CADI targets:

- ARM_Neoverse-V2
- Cluster_ARM_Neoverse-V2
- PVCache
- TlbCadi

About ARMNeoverseV2CT

The model supports the following features:

- DynamIQ™ r3p0.
- DynamIQ™ Shared Unit-110 (DSU-110) system registers.
- L2 cache is supported at the per-core level only and there is no implementation of Core-Complex with shared L2 cache yet.
- PChannel for the cluster and for each core.
- BROADCASTPERSIST pin.
- Optional peripheral port.
- L3Cache partition.
- Per-core clock.
- Utility bus.

Support for the following features is planned for a future release:

- Transporting architectural metadata tags over TLM2 (OSCI) and AMBA-PV (the Arm TLM2 extensions) bus protocols.
- BROADCASTCACHEMAINTPOU pin.
- COREINSTRRET, COREINSTRRUN, and nPMBIRQ signals.
- Core-Complex.
- Each thread currently has its own L1Cache and L2Cache instead of a common cache that is shared by all threads of the core.
- Per-thread parameters are not yet supported but signals are implemented.

The following features will not be implemented:

- DynamIQ features that are irrelevant to the programmers' view simulation, for example:
 - Automatic CPU retention mode.
 - Level-3 Cache RAM retention.

- 256-bit wide output transactions.
- Error correction or detection features.
- Self-test features (MBIST).
- Snoop filtering.
- Latency configuration
- Cache stashing capability
- Embedded Logic Analyzer (ELA).

This model supports the Arm®v8-A Cryptographic Extensions, which requires the crypto plug-in (`crypto.dll` or `crypto.so`) to be loaded. The crypto plug-in is available for download from the [Arm Developer website](#).

The following models also exist, with a fixed number of cores per cluster:

- ARMNeoverseV2x1CT.

Ports for ARMNeoverseV2CT

Table 3-514: Ports

Name	Protocol	Type	Description
acp_s	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[1]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[1]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[1]	Signal	Master	Timer signals to SOC
CNTHVIRQ[1]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[1]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[1]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[1]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[1]	Signal	Master	Timer signals to SOC.
commirq[1]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[1]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[1]	PChannel	Master	Core PCSM signals
core_powerdown_out[1]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[1]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[1]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[1]	Signal	Slave	Disable cryptography extensions after reset.
cti[1]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[1]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[1]	Signal	Master	No power-down request.
dbgprupreq[1]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.

Name	Protocol	Type	Description
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.
fiq[1]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[1]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[1]	Signal	Slave	This signal drives the CPUs interrupt handling.
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[1]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[1]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[1]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[1]	Signal	Slave	PPU Core wake request signals.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[1]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rvbaraddr[1]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[1]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[1]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave
vcpumntirq[1]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.

Name	Protocol	Type	Description
vfiq[1]	Signal	Slave	Virtualised FIQ.
virq[1]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[1]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMNeoverseV2CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpu_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.force-fpsid

Override the FPSID value

Type

bool

Default value

0x1

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_latency

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. l2cache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-size

L2 Cache size in bytes.

Type

int

Default value

0x100000

cpuX.l2cache-snoop_data_transfer_latency

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-snoop_issue_latency

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x8000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x20

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.PIDR_REVAND

TRCPIDR REVAND value

Type

int

Default value

0x0

ete.PIDR_REVISION

TRCPIDR REVISION value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.REVISION

TRCIDR1 revision value

Type

int

Default value

0x0

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

ext_abort_normal_noncacheable_read_is_sync

Synchronous reporting of normal noncacheable-read external aborts

Type

bool

Default value

0x1

ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x2

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_coherent_icache

Whether icache invalidation to the point of unification is required for instruction to data coherence. true - Invalidate operations not required

Type

bool

Default value

0x1

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_external_rndr

Implement external random number generator module. When enabling this with has_rndr enabled, the external random number generator will be used instead of internal random number generator

Type

int

Default value

0x1

has_large_va

Implement support for the extended 52-bit virtual addresses from ARMv8.2 (FEAT_LVA).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.2 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_rndr

Implement random number instructions to read from RNDR and RNDRSS random number registers from ARMv8.5 (FEAT_RNG).

Type

int

Default value

0x1

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or

miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

`icache-read_latency`

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

`icache-size`

L1 I-Cache size in bytes.

Type

int

Default value

0x8000

`icache-state_modelled`

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

`instruction_tlb_size`

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

`memory_tagging_support_level`

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpam_max_partid

MPAM Maximum PARTID Supported

Type

int

Default value

0x1ff

mpam_max_vpmr

MPAM Maximum VPMR Supported

Type

int

Default value

0x7

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use (n * accumulator value) to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantums in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

pmu-num_counters

Number of PMU counters implemented

Type

int

Default value

0x6

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

rndr_rndrrs_seed

Initial seed for random engine used in RNDR register

Type

int

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

trace_physical_registers_when_host_virtualisation_enabled

When host virtualisation is enabled, trace sysreg accesses to physical register accessed (0=disabled, 1=Trace only ELR/SPSR_EL1 as ELR/SPSR_EL2, 2=Trace all redirected registers as physical registers)

Type

int

Default value

0x1

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.81 ARMNeoverseV3AECT

ARMNeoverseV3AECT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-515: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Ports added:

- legacy_tz_en

Parameters added:

- mpmm_accumulator_multiplier
- num_acp
- tlbi_stall_enabled

Iris and MTI instances for ARMNeoverseV3AECT

This model has the following Iris instances:

Table 3-516: ARMNeoverseV3AECT Iris instances

InstanceName	ComponentName
ARMNeoverseV3AECT	Cluster_ARM_Neoverse-V3AE
ARMNeoverseV3AECT.AMU	PVBusLogger
ARMNeoverseV3AECT.AMU.mapper	PVBusMapper
ARMNeoverseV3AECT.DAP	PVBusLogger
ARMNeoverseV3AECT.DAP.mapper	PVBusMapper
ARMNeoverseV3AECT.DSU	DSU
ARMNeoverseV3AECT.DSU.PPU_cluster	PPUv1
ARMNeoverseV3AECT.DSU.PPU_cluster.busslave	PVBusSlave
ARMNeoverseV3AECT.DSU.PPU_core0	PPUv1
ARMNeoverseV3AECT.DSU.PPU_core0.busslave	PVBusSlave
ARMNeoverseV3AECT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseV3AECT.DSU.shared_cache	PVCache
ARMNeoverseV3AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseV3AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseV3AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseV3AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseV3AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMNeoverseV3AECT.DSU.utility_slave[0]	PVBusSlave
ARMNeoverseV3AECT.MMAP	PVBusLogger
ARMNeoverseV3AECT.MMAP.mapper	PVBusMapper
ARMNeoverseV3AECT.cpu0	ARM_Neoverse-V3AE
ARMNeoverseV3AECT.cpu0.UTLB	TLB
ARMNeoverseV3AECT.cpu0.dtlb	TlbCadi
ARMNeoverseV3AECT.cpu0.l1dcache	PVCache
ARMNeoverseV3AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseV3AECT.cpu0.l1icache	PVCache
ARMNeoverseV3AECT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseV3AECT.cpu0.l2cache	PVCache
ARMNeoverseV3AECT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseV3AECT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseV3AECT.ext_bus	PVBusLogger
ARMNeoverseV3AECT.ext_bus.mapper	PVBusMapper
ARMNeoverseV3AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-517: ARMNeoverseV3AECT MTI instances

InstanceName	ComponentName
ARMNeoverseV3AECT	ARMv8Cluster

InstanceName	ComponentName
ARMNeoverseV3AECT.AMU	PVBusLogger
ARMNeoverseV3AECT.AMU.mapper	PVBusMapper
ARMNeoverseV3AECT.DAP	PVBusLogger
ARMNeoverseV3AECT.DAP.mapper	PVBusMapper
ARMNeoverseV3AECT.DSU	DSU
ARMNeoverseV3AECT.DSU.PPU_cluster	PPUv1
ARMNeoverseV3AECT.DSU.PPU_cluster.busslave	PVBusSlave
ARMNeoverseV3AECT.DSU.PPU_core0	PPUv1
ARMNeoverseV3AECT.DSU.PPU_core0.busslave	PVBusSlave
ARMNeoverseV3AECT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseV3AECT.DSU.shared_cache	PVCache
ARMNeoverseV3AECT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseV3AECT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseV3AECT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseV3AECT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseV3AECT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMNeoverseV3AECT.DSU.utility_slave[0]	PVBusSlave
ARMNeoverseV3AECT.MMAP	PVBusLogger
ARMNeoverseV3AECT.MMAP.mapper	PVBusMapper
ARMNeoverseV3AECT.cpu0	ARM_Neoverse-V3AE
ARMNeoverseV3AECT.cpu0.UTLB	TLB
ARMNeoverseV3AECT.cpu0.l1dcache	PVCache
ARMNeoverseV3AECT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseV3AECT.cpu0.l1icache	PVCache
ARMNeoverseV3AECT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseV3AECT.cpu0.l2cache	PVCache
ARMNeoverseV3AECT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseV3AECT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseV3AECT.ext_bus	PVBusLogger
ARMNeoverseV3AECT.ext_bus.mapper	PVBusMapper
ARMNeoverseV3AECT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMNeoverseV3AECT contains the following CADI targets:

- ARM_Neoverse-V3AE
- Cluster_ARM_Neoverse-V3AE
- PVCache
- TlbCadi

AE-specific features implemented

The only AE-specific features implemented in the model are Split-Lock and Lock-Step, with the following limitations:

- The implementation of Split-Lock and Lock-Step in the model is simply a halving of the number of cores available.
- Lock-step is implemented by a parameter that can be true or false for the entire execution only, rather than by a signal that can be enabled during execution.

As the model does not implement true Split-Lock functionality, the ports and parameters are contiguous. For example, for a 4-core CPU with Split-Lock enabled, `cpu0` and `cpu1` identify the available cores and associated ports, not `cpu0` and `cpu2`.

- Hybrid mode is not modeled in the DSU.

Ports for ARMNeoverseV3AECT

Table 3-518: Ports

Name	Protocol	Type	Description
<code>acp_s[2]</code>	PVBus	Slave	AXI ACP slave port.
<code>AEND0MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND1MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND2MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>AEND3MP</code>	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
<code>ASTART0MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART1MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART2MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>ASTART3MP</code>	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
<code>broadcastatomic</code>	Signal	Slave	CHI defined pins.
<code>broadcastcachemaint</code>	Signal	Slave	ACE defined pins.
<code>broadcastouter</code>	Signal	Slave	ACE defined pins.
<code>broadcastpersist</code>	Signal	Slave	CHI defined pins.
<code>cfgend[1]</code>	Signal	Slave	This signal is for EE bit initialisation.
<code>cfgsdisable</code>	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
<code>clk_in</code>	ClockSignal	Slave	The clock signal connected to the <code>clk_in</code> port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.

Name	Protocol	Type	Description
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[1]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[1]	Signal	Master	Timer signals to SOC
CNTHVIRQ[1]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[1]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[1]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[1]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[1]	Signal	Master	Timer signals to SOC.
commirq[1]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[1]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[1]	PChannel	Master	Core PCSM signals
core_powerdown_out[1]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[1]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[1]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[1]	Signal	Slave	Disable cryptography extensions after reset.
cti[1]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[1]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[1]	Signal	Master	No power-down request.
dbgpwrupreq[1]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.

Name	Protocol	Type	Description
fiq[1]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[1]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[1]	Signal	Slave	This signal drives the CPUs interrupt handling.
l0gptsz	Value	Slave	RME LOGPTSZ port
legacy_tz_en	Signal	Slave	RME LEGACY_TZ_EN port
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[1]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[1]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[1]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[1]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[1]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rlpiden	Signal	Slave	External debug interface.
rtpiden	Signal	Slave	External debug interface.
rvbaraddr[1]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[1]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[1]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave

Name	Protocol	Type	Description
vcpumntirq[1]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[1]	Signal	Slave	Virtualised FIQ.
virq[1]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[1]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMNeoverseV3AECT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpu_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.crypto_aes

AES instructions supported (requires CryptoPlugin to be loaded). 0, not implemented. 2, AES and PMULL instructions implemented (FEAT_AES, FEAT_PMULL).

Type

int

Default value

0x2

cpuX.crypto_sha3

Implement ARMv8.4 SHA-3 instructions (requires CryptoPlugin to be loaded) (FEAT_SHA3).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

cpuX.crypto_sha512

Implement ARMv8.4 SHA-512 instructions (requires CryptoPlugin to be loaded) (FEAT_SHA512).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

cpuX.crypto_sm3

Implement ARMv8.4 SM-3 instructions (requires CryptoPlugin to be loaded) (FEAT_SM3).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

cpuX.crypto_sm4

Implement ARMv8.4 SM-4 instructions (requires CryptoPlugin to be loaded) (FEAT_SM4).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-read_latency`

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-size`

L2 Cache size in bytes.

Type

int

Default value

0x80000

`cpuX.l2cache-snoop_data_transfer_latency`

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-snoop_issue_latency`

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-ways`

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

ecv_support_level

Implement Enhanced Counter Virtualization feature from ARMv8.6. 0, Not supported. 1, fully supported without CNTPOFF. 2, fully supported with CNTPOFF (FEAT_ECV).

Type

int

Default value

0x2

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x2

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_rndr

Implement random number instructions to read from RNDR and RNDRSS random number registers from ARMv8.5 (FEAT_RNG).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

has_v8_7_spe_inverted_filtering

Where FEAT_SPEv1p2 is implemented, whether the inverting filtering by events

Type

bool

Default value

0x0

has_v8_7_spe_previous_branch_target

Where FEAT_SPEv1p2 is implemented, whether the optional branch target feature is implemented

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

log2_trace_buffer_alignment

Log2 of trace buffer alignment constraint for output buffer (0->1B ... 11->2Kib)

Type

int

Default value

0x6

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpamidr_has_force_ns

Whether MPAMIDR_EL1.HAS_FORCE_NS bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

mpamidr_has_sdeflt

Whether MPAMIDR_EL1.HAS_SDEFLT bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

mpamidr_has_tidr

Whether MPAMIDR_EL1.HAS_TIDR bit is set or clear

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use $(n * \text{accumulator value})$ to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_acp

Number of ACP ports

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

rme_level0_gpt_size

The range of address space protected by each entry in the level 0 GPT (0->1GB 1->16GB, 2->64GB, 3->512GB)

Type

int

Default value

0x0

rme_support_level

0 -> Realm management extension not implemented, 1 -> LEGACY_TZ_EN mode i.e. RME register fields are stateful but only supports secure/non-secure states, 2 -> Realm management extension fully implemented (FEAT_RME).

Type

int

Default value

0x2

rndr_rndrrs_seed

Initial seed for random engine used in RNDR register

Type

int

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tcr_txsz_undersize_should_fault

If large VA is not supported, Whether undersized TxSZ value should generate translation fault

Type

bool

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.82 ARMNeoverseV3CT

ARMNeoverseV3CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-519: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Ports added:

- cluster_pcs_m_pchannel
- core_pcs_m_pchannel

Parameters added:

- mpm_accumulator_multiplier
- tlbi_stall_enabled

Iris and MTI instances for ARMNeoverseV3CT

This model has the following Iris instances:

Table 3-520: ARMNeoverseV3CT Iris instances

InstanceName	ComponentName
ARMNeoverseV3CT	Cluster_ARM_Neoverse-V3
ARMNeoverseV3CT.AMU	PVBusLogger
ARMNeoverseV3CT.AMU.mapper	PVBusMapper
ARMNeoverseV3CT.DAP	PVBusLogger
ARMNeoverseV3CT.DAP.mapper	PVBusMapper
ARMNeoverseV3CT.DSU	DSU
ARMNeoverseV3CT.DSU.PPU_cluster	PPUv1
ARMNeoverseV3CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMNeoverseV3CT.DSU.PPU_core0	PPUv1
ARMNeoverseV3CT.DSU.PPU_core0.busslave	PVBusSlave
ARMNeoverseV3CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseV3CT.DSU.shared_cache	PVCache
ARMNeoverseV3CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseV3CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseV3CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseV3CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseV3CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMNeoverseV3CT.DSU.utility_slave[0]	PVBusSlave
ARMNeoverseV3CT.MMAP	PVBusLogger
ARMNeoverseV3CT.MMAP.mapper	PVBusMapper
ARMNeoverseV3CT.cpu0	ARM_Neoverse-V3
ARMNeoverseV3CT.cpu0.UTLB	TLB
ARMNeoverseV3CT.cpu0.dtlb	TlbCadi
ARMNeoverseV3CT.cpu0.l1dcache	PVCache
ARMNeoverseV3CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseV3CT.cpu0.l1icache	PVCache

InstanceName	ComponentName
ARMNeoverseV3CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseV3CT.cpu0.l2cache	PVCache
ARMNeoverseV3CT.cpu0.l2cache.upstream[0]	PVBusSlave
ARMNeoverseV3CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseV3CT.ext_bus	PVBusLogger
ARMNeoverseV3CT.ext_bus.mapper	PVBusMapper
ARMNeoverseV3CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

This model has the following MTI trace components:

Table 3-521: ARMNeoverseV3CT MTI instances

InstanceName	ComponentName
ARMNeoverseV3CT	ARMv8Cluster
ARMNeoverseV3CT.AMU	PVBusLogger
ARMNeoverseV3CT.AMU.mapper	PVBusMapper
ARMNeoverseV3CT.DAP	PVBusLogger
ARMNeoverseV3CT.DAP.mapper	PVBusMapper
ARMNeoverseV3CT.DSU	DSU
ARMNeoverseV3CT.DSU.PPU_cluster	PPUv1
ARMNeoverseV3CT.DSU.PPU_cluster.busslave	PVBusSlave
ARMNeoverseV3CT.DSU.PPU_core0	PPUv1
ARMNeoverseV3CT.DSU.PPU_core0.busslave	PVBusSlave
ARMNeoverseV3CT.DSU.mpam_busslave	PVBusSlave
ARMNeoverseV3CT.DSU.shared_cache	PVCache
ARMNeoverseV3CT.DSU.shared_cache.upstream[0]	PVBusSlave
ARMNeoverseV3CT.DSU.shared_cache.upstream[1]	PVBusSlave
ARMNeoverseV3CT.DSU.shared_cache.upstream[2]	PVBusSlave
ARMNeoverseV3CT.DSU.shared_cache.upstream[3]	PVBusSlave
ARMNeoverseV3CT.DSU.shared_cache.upstream[4]	PVBusSlave
ARMNeoverseV3CT.DSU.utility_slave[0]	PVBusSlave
ARMNeoverseV3CT.MMAP	PVBusLogger
ARMNeoverseV3CT.MMAP.mapper	PVBusMapper
ARMNeoverseV3CT.cpu0	ARM_Neoverse-V3
ARMNeoverseV3CT.cpu0.UTLB	TLB
ARMNeoverseV3CT.cpu0.l1dcache	PVCache
ARMNeoverseV3CT.cpu0.l1dcache.upstream[0]	PVBusSlave
ARMNeoverseV3CT.cpu0.l1icache	PVCache
ARMNeoverseV3CT.cpu0.l1icache.upstream[0]	PVBusSlave
ARMNeoverseV3CT.cpu0.l2cache	PVCache
ARMNeoverseV3CT.cpu0.l2cache.upstream[0]	PVBusSlave

InstanceName	ComponentName
ARMNeoverseV3CT.cpu0.l2cache.upstream[1]	PVBusSlave
ARMNeoverseV3CT.ext_bus	PVBusLogger
ARMNeoverseV3CT.ext_bus.mapper	PVBusMapper
ARMNeoverseV3CT.gic_cpuif_decoder_cluster	GICv3CPUInterfaceDecoder

ARMNeoverseV3CT contains the following CADI targets:

- ARM_Neoverse-V3
- Cluster_ARM_Neoverse-V3
- PVCache
- TlbCadi

Ports for ARMNeoverseV3CT

Table 3-522: Ports

Name	Protocol	Type	Description
acp_s[2]	PVBus	Slave	AXI ACP slave port.
AEND0MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND1MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND2MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
AEND3MP	Value_64	Slave	Port to obtain end address of valid peripheral address range (exclusive).
ASTART0MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART1MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART2MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
ASTART3MP	Value_64	Slave	Port to obtain start address of valid peripheral address range (inclusive).
broadcastatomic	Signal	Slave	CHI defined pins.
broadcastcachemaint	Signal	Slave	ACE defined pins.
broadcastouter	Signal	Slave	ACE defined pins.
broadcastpersist	Signal	Slave	CHI defined pins.
cfgend[1]	Signal	Slave	This signal is for EE bit initialisation.
cfgsdisable	Signal	Slave	This signal disables write access to some secure Interrupt Controller registers.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the cluster level components run e.g cluster level timers, caches and pmu.

Name	Protocol	Type	Description
cluster_pcs_m_pchannel	PChannel	Master	Cluster PCSM signal
cluster_powerdown_out	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the cluster power info
clusterid	Value	Slave	The port reads the value in CPU ID register field, bits[11:8] of the MPIDR.
clusterpmuirq	Signal	Master	DynamlQ pmu irq
CNTHPIRQ[1]	Signal	Master	Timer signals to SOC.
CNTHPSIRQ[1]	Signal	Master	Timer signals to SOC
CNTHVIRQ[1]	Signal	Master	Timer signals to SOC.
CNTHVSIRQ[1]	Signal	Master	Timer signals to SOC
CNTPNSIRQ[1]	Signal	Master	Timer signals to SOC.
CNTPSIRQ[1]	Signal	Master	Timer signals to SOC.
cntvalueb	CounterInterface	Slave	Interface to SoC level counter module.
CNTVIRQ[1]	Signal	Master	Timer signals to SOC.
commirq[1]	Signal	Master	Interrupt signal from debug communications channel.
core_clk_in[1]	ClockSignal	Slave	The clock signal connected to the core_clk_in port is used to determine the rate at which each core executes instructions.
core_pcs_m_pchannel[1]	PChannel	Master	Core PCSM signals
core_powerdown_out[1]	Signal	Master	DEPRECATED - An output from PPU that informs thermal controller of the core power info
coreerrirq[1]	Signal	Master	Error indicator for an ECC error that causes potential data corruption or loss of coherency
corefaultirq[1]	Signal	Master	Fault indicator for a detected 1 or 2 bit ECC error
cryptodisable[1]	Signal	Slave	Disable cryptography extensions after reset.
cti[1]	v8EmbeddedCrossTrigger_controlprotocol	Master	Cross trigger matrix port.
ctidbgirq[1]	Signal	Master	Cross Trigger Interface (CTI) interrupt trigger output.
dbgen	Signal	Slave	External debug interface.
dbgnopwrdown[1]	Signal	Master	No power-down request.
dbgpwrupreq[1]	Signal	Master	Debug power up request.
defaultMP	Signal	Slave	If DefaultMP is asserted, all the transactions go to peripheral ports except the configured address ranges. Configured address ranges are sent to main master interface ports.
dev_debug_s	PVBus	Slave	External debug interface.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE.

Name	Protocol	Type	Description
fiq[1]	Signal	Slave	This signal drives the CPUs fast-interrupt handling.
gicreset	Signal	Master	An output from PPU that can be used to reset GICCLK domain components
gicv3_redistributor_s[1]	GICv3Comms	Slave	GICv3 AXI-stream port.
irq[1]	Signal	Slave	This signal drives the CPUs interrupt handling.
l0gptsz	Value	Slave	RME LOGPTSZ port
legacy_tz_en	Signal	Slave	RME LEGACY_TZ_EN port
memorymapped_debug_s	PVBus	Slave	External debug interface.
pmbirq[1]	Signal	Master	Interrupt signal from the statistical profiling unit.
pmuirq[1]	Signal	Master	Interrupt signal from performance monitoring unit.
ppu_cluster_irq	Signal	Master	PPU cluster interrupt.
ppu_cluster_wakerequest	Signal	Slave	PPU cluster wake request signal.
ppu_core_irq[1]	Signal	Master	PPU core interrupt.
ppu_core_wakerequest[1]	Signal	Slave	PPU core wake request signal.
presetdbg	Signal	Slave	Initialize the shared debug APB, Cross Trigger Interface (CTI), and Cross Trigger Matrix (CTM) logic.
pvbus_m0	PVBus	Master	The core will generate bus requests on this port.
pvbus_periph_m	PVBus	Master	The core can generate peripheral bus request on this port.
rei[1]	Signal	Slave	Per core RAM Error Interrupt.
reset	Signal	Slave	Reset signal to cluster.
rlpiden	Signal	Slave	External debug interface.
rtpiden	Signal	Slave	External debug interface.
rvbaraddr[1]	Value_64	Slave	Reset vector base address.
sci_m	SystemCoherencyInterface	Master	System coherency interface port, which is used to take the whole cluster into/out-of coherency domain
sei[1]	Signal	Slave	Per core System Error physical pins.
spiden	Signal	Slave	External debug interface.
ticks[1]	InstructionCount	Master	This port should be connected to one of the two ticks ports on a 'visualisation' component, in order to display a running instruction count.
trbirq[1]	Signal	Master	Interrupt signal from the trace buffer unit.
utility_bus_s	PVBus	Slave	Utility bus slave

Name	Protocol	Type	Description
vcpumntirq[1]	Signal	Master	Interrupt signal for virtual CPU maintenance IRQ.
vfiq[1]	Signal	Slave	Virtualised FIQ.
virq[1]	Signal	Slave	Virtualised IRQ.
virtio_s	PVBus	Slave	The virtio coherent port, hooks directly into the L2 system and becomes coherent (assuming attributes are set correctly).
vsei[1]	Signal	Slave	Per core virtual System Error physical pins.

Parameters for ARMNeoverseV3CT

AEND0_DEFAULT

Default end address for peripheral port 0 address range exclusive (corresponds to AENDMP0 input signal).

Type

int

Default value

0x0

AEND1_DEFAULT

Default end address for peripheral port 1 address range exclusive (corresponds to AENDMP1 input signal).

Type

int

Default value

0x0

AEND2_DEFAULT

Default end address for peripheral port 2 address range exclusive (corresponds to AENDMP2 input signal).

Type

int

Default value

0x0

AEND3_DEFAULT

Default end address for peripheral port 3 address range exclusive (corresponds to AENDMP3 input signal).

Type

int

Default value

0x0

ASTART0_DEFAULT

Default start address for peripheral port 0 address range inclusive (corresponds to ASTARTMP0 input signal).

Type

int

Default value

0x0

ASTART1_DEFAULT

Default start address for peripheral port 1 address range inclusive (corresponds to ASTARTMP1 input signal).

Type

int

Default value

0x0

ASTART2_DEFAULT

Default start address for peripheral port 2 address range inclusive (corresponds to ASTARTMP2 input signal).

Type

int

Default value

0x0

ASTART3_DEFAULT

Default start address for peripheral port 3 address range inclusive (corresponds to ASTARTMP3 input signal).

Type

int

Default value

0x0

BROADCASTATOMIC

Enable broadcasting of atomic operation. The broadcastatomic signal will override this value if used

Type

bool

Default value

0x1

BROADCASTCACHEMAINT

Enable broadcasting of cache maintenance operations to downstream caches. The broadcastcachemaint signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTOUTER

Enable broadcasting of Outer Shareable transactions. The broadcastouter signal will override this value if used.

Type

bool

Default value

0x0

BROADCASTPERSIST

Enable broadcasting of cache clean to the point of persistence operations. The broadcastpersist signal will override this value if used

Type

bool

Default value

0x1

CLUSTER_ID

Processor cluster ID value

Type

int

Default value

0x0

CMO_broadcast_when_cache_state_modelling_disabled

Skip broadcasting some cache maintenance operations (CMOs) when cache state modelling is disabled as a performance optimisation. Changing this value may affect performance of the simulation, but is required if the rest of the system is sensitive to certain CMOs even when cache state modelling is disabled. Possible values are: - 0 = All CMOs are broadcast if architecturally required - 1 = Data cache maintenance to the PoC or PoU are not broadcast when data cache state modelling is disabled. Otherwise all CMOs are broadcast if architecturally required

Type

int

Default value

0x1

CPUCFR

Value of CPU Configuration Register

Type

int

Default value

0x0

GICDISABLE

Disable the new style GICv3 CPU interface in each core model. It can be set for the platform not having a GICv3 Interface (GICv3)

Type

bool

Default value

0x1

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x1

bus_type

Cosmetic change that changes reset value of L2ACTLR register. 0, ACE. 1, CHI. 2, AXI

Type

int

Default value

0x0

core_power_on_by_default

If true, The cluster and cores will be powered on for v9 after reset. If this parameter is enabled, the powerdown signal from in-cluster PPU which is connected to cores and cluster reset signals will be cleared after reset and triggers the power on sequence in PPU

Type

bool

Default value

0x0

cpu_div

Divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpu_mul

Multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpuX.CFGEND

Endianness configuration at reset. 0, little endian. 1, big endian.

Type

bool

Default value

0x0

cpuX.CFGTE

Instruction set state when resetting into AArch32. 0, A32. 1, T32.

Type

bool

Default value

0x0

cpuX.CRYPTODISABLE

Disable cryptographic features.

Type

bool

Default value

0x0

cpuX.RVBARADDR

Value of RVBAR_ELx register.

Type

int

Default value

0x0

cpuX.crypto_aes

AES instructions supported (requires CryptoPlugin to be loaded). 0, not implemented. 2, AES and PMULL instructions implemented (FEAT_AES, FEAT_PMULL).

Type

int

Default value

0x2

cpuX.crypto_sha3

Implement ARMv8.4 SHA-3 instructions (requires CryptoPlugin to be loaded) (FEAT_SHA3).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

cpuX.crypto_sha512

Implement ARMv8.4 SHA-512 instructions (requires CryptoPlugin to be loaded) (FEAT_SHA512).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

cpuX.crypto_sm3

Implement ARMv8.4 SM-3 instructions (requires CryptoPlugin to be loaded) (FEAT_SM3).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

cpuX.crypto_sm4

Implement ARMv8.4 SM-4 instructions (requires CryptoPlugin to be loaded) (FEAT_SM4).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.4 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

cpuX.enable_trace_special_hlt_imm16

Enable usage of parameter trace_special_hlt_imm16

Type

bool

Default value

0x0

cpuX.l2cache-hit_latency

L2 Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-maintenance_latency

L2 Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-miss_latency

L2 Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-read_access_latency

L2 Cache timing annotation latency for read accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond

to the time taken to transfer across the cache upstream bus, this is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-read_latency`

L2 Cache timing annotation latency for read accesses given in ticks per byte accessed. `l2cache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-size`

L2 Cache size in bytes.

Type

int

Default value

0x200000

`cpuX.l2cache-snoop_data_transfer_latency`

L2 Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-snoop_issue_latency`

L2 Cache timing annotation latency for snoop accesses issued by this cache in total ticks. This is only used when `dcache-state_modelled=true`.

Type

int

Default value

0x0

`cpuX.l2cache-ways`

L2 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x8

cpuX.l2cache-write_access_latency

L2 Cache timing annotation latency for write accesses given in ticks per access. If this parameter is non-zero, per-access latencies will be used instead of per-byte even if l2cache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.l2cache-write_latency

L2 Cache timing annotation latency for write accesses given in ticks per byte accessed. l2cache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

cpuX.max_code_cache_mb

Maximum size of the simulation code cache (MiB). For platforms with more than 2 cores this limit will be scaled down. (e.g 1/8 for 16 or more cores)

Type

int

Default value

0x100

cpuX.min_sync_level

Force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

cpuX.semihosting-A32_HLT

A32 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-A64_HLT

A64 HLT number for semihosting calls.

Type

int

Default value

0xf000

cpuX.semihosting-ARM_SVC

A32 SVC number for semihosting calls.

Type

int

Default value

0x123456

cpuX.semihosting-T32_HLT

T32 HLT number for semihosting calls.

Type

int

Default value

0x3c

cpuX.semihosting-Thumb_SVC

T32 SVC number for semihosting calls.

Type

int

Default value

0xab

cpuX.semihosting-cmd_line

Command line available to semihosting calls.

Type

string

Default value**cpuX.semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**cpuX.semihosting-enable**

Enable semihosting SVC/HLT traps.

Type

bool

Default value

0x1

cpuX.semihosting-heap_base

Virtual address of heap base.

Type

int

Default value

0x0

cpuX.semihosting-heap_limit

Virtual address of top of heap.

Type

int

Default value

0xf000000

cpuX.semihosting-stack_base

Virtual address of base of descending stack.

Type

int

Default value

0x10000000

cpuX.semihosting-stack_limit

Virtual address of stack limit.

Type

int

Default value

0xf000000

cpuX.trace_special_hlt_imm16

For this HLT number, IF enable_trace_special_hlt_imm16=true, skip performing usual HLT execution but call MTI trace if registered

Type

int

Default value

0xf000

cpuX.vfp-enable_at_reset

Enable VFP in CPACR, CPPWR, NSACR at reset. Warning: Arm recommends going through the implementation's suggested VFP power-up sequence!

Type

bool

Default value

0x0

cpuX.vfp-present

Set whether the model has VFP support

Type

bool

Default value

0x1

dcache-hit_latency

L1 D-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-maintenance_latency

L1 D-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-miss_latency

L1 D-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-prefetch_enabled

Enable simulation of data cache prefetching. This is only used when dcache-state_modelled=true

Type

bool

Default value

0x0

dcache-read_access_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per access (of size dcache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-read_latency

L1 D-Cache timing annotation latency for read accesses given in ticks per byte accessed. dcache-read_access_latency must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-size

L1 D-Cache size in bytes.

Type

int

Default value

0x10000

dcache-snoop_data_transfer_latency

L1 D-Cache timing annotation latency for received snoop accesses that perform a data transfer given in ticks per byte accessed. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-state_modelled

Set whether D-cache has stateful implementation

Type

bool

Default value

0x0

dcache-write_access_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per access (of size dcache-write_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if dcache-write_latency is set. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

dcache-write_latency

L1 D-Cache timing annotation latency for write accesses given in ticks per byte accessed. dcache-write_access_latency must be set to 0 for per-byte latencies to be applied. This is only used when dcache-state_modelled=true.

Type

int

Default value

0x0

default_opmode

Operating mode of DynamIQ coming out of reset. 0: SFONLY ON, 1: 1/4 CACHE ON, 2: 1/2 CACHE ON, 3: 3/4 CACHE ON, 4: FULL CACHE ON

Type

int

Default value

0x4

diagnostics

Enable DynamIQ diagnostic messages

Type

bool

Default value

0x0

ecv_support_level

Implement Enhanced Counter Virtualization feature from ARMv8.6. 0, Not supported. 1, fully supported without CNTPOFF. 2, fully supported with CNTPOFF (FEAT_ECV).

Type

int

Default value

0x2

enable_simulation_performance_optimizations

With this option enabled, the model will run more quickly, but be less accurate to exact CPU behavior. The model will still be functionally accurate for software, but may increase differences seen between hardware behavior and model behavior for certain workloads (it changes the micro-architectural value of stage12_tlb_size parameter to 1024).

Type

bool

Default value

0x1

ete.CLAIMTAGS

Number of claim tags

Type

int

Default value

0x4

ete.MAX_INST_PER_Q

Maximum limit for the number of instructions implied by a Q element

Type

int

Default value

0x1

ete.NumberOfRSPairs

Number of resource selector pairs

Type

int

Default value

0x8

ete.PIDR_CM0D

TRCPIDR CM0D value

Type

int

Default value

0x0

ete.Q_CADENCE

Number of instruction blocks traced between two Q elements

Type

int

Default value

0x1

ete.RES0_STATEFUL

Whether RES0 bits are stateful or RAZ/WI

Type

bool

Default value

0x0

ete.RETSTACK

Return stack depth

Type

int

Default value

0x3

ete.SIM_OVERFLOW_GRANULARITY

Number of instruction blocks in each granule, for simulated overflow

Type

int

Default value

0x64

ete.SIM_OVERFLOW_PERCENTAGE

Percentage of instruction blocks lost in each granule, for simulated overflow

Type

int

Default value

0x0

ete.SOURCE_ADDRESS

Allow generation of source address elements

Type

bool

Default value

0x0

ete.TRACE_OUTPUT

File to which to write trace byte stream

Type

string

Default value**ete.TRCRSRTA_FORCED_EXCEP**

TRCRSR.TA value for a forcibly traced exception

Type

bool

Default value

0x0

ext_abort_so_write_ras_type

External Aborts are reported as RAS error type specified in this param. Values: 0 = NONE, 1 = UC, 2 = UEU, 3 = UEO, 4 = UER, 5 = CE.

Type

int

Default value

0x2

force_mte_tag_access_razwi_and_ignore_tag_checks

Force MTE tag accesses to RAZ/WI and also ignore tag checks irrespective of whether memory is tagged or not. This parameter doesn't affect non-load/store tag generation instructions like ADDG/GMI/IRG/SUBG. Please note that setting this parameter to true will also disable MTE tag access and tag check related traces

Type

bool

Default value

0x0

force_zero_PSTATE_PAN

Non-architecture parameter to force PSTATE.PAN to be 0.0: No effect. 1: PSTATE.PAN is always treated as 0. The parameter optimizes the performance of updating PSTATE.PAN.

Type

bool

Default value

0x0

force_zero_mpam_partid_and_pmg

Non-architecture parameter to force MPAM PARTID, PMG and NS/SP to 0.0: No effect. 1: PARTID and PMG are always treated as 0. The parameter optimizes the performance of updating the registers MPAM0_EL1, MPAM1_EL1, MPAM2_EL2, MPAM3_EL3, MPAMHCR_EL2.

Type

int

Default value

0x0

has_enhanced_pan

Implements Armv8.7 Enhanced PAN feature (FEAT_PAN3)

Possible values of this parameter are: - 1, feature is implemented if ARMv8.7 is enabled. - 2, feature is implemented.

Type

int

Default value

0x2

has_ete

If true, implements the Embedded Trace Extension (FEAT_ETE). This option is discarded if ete plugin is explicitly loaded (--plugin or -P)

Type

bool

Default value

0x0

has_peripheral_port

If true, an additional AXI peripheral port is configured.

Type

bool

Default value

0x0

has_rndr

Implement random number instructions to read from RNDR and RNDRSS random number registers from ARMv8.5 (FEAT_RNG).

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.5 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

has_statistical_profiling

Whether Statistical Based Profiling is implemented (FEAT_SPE).

Type

bool

Default value

0x1

has_v8_7_spe_inverted_filtering

Where FEAT_SPEv1p2 is implemented, whether the inverting filtering by events

Type

bool

Default value

0x0

has_v8_7_spe_previous_branch_target

Where FEAT_SPEv1p2 is implemented, whether the optional branch target feature is implemented

Type

bool

Default value

0x0

icache-hit_latency

L1 I-Cache timing annotation latency for hit. Intended to model the tag-lookup time. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-maintenance_latency

L1 I-Cache timing annotation latency for cache maintenance operations given in total ticks. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-miss_latency

L1 I-Cache timing annotation latency for miss. Intended to model the time for failed tag-lookup and allocation of intermediate buffers. This is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-prefetch_enabled

Enable simulation of instruction cache prefetching. This is only used when icache-state_modelled=true.

Type

bool

Default value

0x0

icache-read_access_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per access (of size icache-read_bus_width_in_bytes). If this parameter is non-zero, per-access latencies will be used instead of per-byte even if icache-read_latency is set. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus, this is only used when icache-state_modelled=true.

Type

int

Default value

0x0

icache-read_latency

L1 I-Cache timing annotation latency for read accesses given in ticks per byte accessed. `icache-read_access_latency` must be set to 0 for per-byte latencies to be applied. This is in addition to the hit or miss latency, and intended to correspond to the time taken to transfer across the cache upstream bus. This is only used when `icache-state_modelled=true`.

Type

int

Default value

0x0

icache-size

L1 I-Cache size in bytes.

Type

int

Default value

0x10000

icache-state_modelled

Set whether I-cache has stateful implementation

Type

bool

Default value

0x0

instruction_tlb_size

Number of stage1+2 itlb entries (or 0 for unified ITLB+DTLB)

Type

int

Default value

0x0

l3cache-ways

L3 Cache number of ways (sets are implicit from size).

Type

int

Default value

0x10

log2_trace_buffer_alignment

Log2 of trace buffer alignment constraint for output buffer (0->1B ... 11->2Kib)

Type

int

Default value

0x6

memory_tagging_support_level

Specify the memory tagging extension support level: 0, not implemented.1, instructions and registers only are implemented (FEAT_MTE).2, implemented (FEAT_MTE2).3, implemented with asymmetric handling of exceptions (FEAT_MTE3).

Type

int

Default value

0x3

mpamidr_has_force_ns

Whether MPAMIDR_EL1.HAS_FORCE_NS bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x0

mpamidr_has_sdeflt

Whether MPAMIDR_EL1.HAS_SDEFLT bit is set or clear.

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

mpamidr_has_tidr

Whether MPAMIDR_EL1.HAS_TIDR bit is set or clear

Possible values of this parameter are: - 0, feature is not enabled. - 1, feature is implemented if ARMv8.6 is enabled. - 2, feature is implemented.

Type

int

Default value

0x1

mpmm_accumulator_multiplier

Parameter for non-architectural behaviour to provide a multiplier for the MPMM accumulator. A value of n means the accumulator will use $(n * \text{accumulator value})$ to calculate the mpmm threshold (MPMM). This is provided as a fast model workaround to handle cases where execution of quantum in SystemC does not result in an expected threshold in MPMM counters being reached at the anticipated time.

Type

int

Default value

0x1

num_acp

Number of ACP ports

Type

int

Default value

0x0

ptw_latency

Page table walker latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the ID registers. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

rme_level0_gpt_size

The range of address space protected by each entry in the level 0 GPT (0->1GB 1->16GB, 2->64GB, 3->512GB)

Type

int

Default value

0x0

rme_support_level

0 -> Realm management extension not implemented, 1 -> LEGACY_TZ_EN mode i.e. RME register fields are stateful but only supports secure/non-secure states, 2 -> Realm management extension fully implemented (FEAT_RME).

Type

int

Default value

0x2

rndr_rndrrs_seed

Initial seed for random engine used in RNDR register

Type

int

Default value

0x0

stage12_tlb_size

Number of stage1+2 tlb entries.

Type

int

Default value

0x80

tcr_txsz_undersize_should_fault

If large VA is not supported, Whether undersized TxSZ value should generate translation fault

Type

bool

Default value

0x0

tlb_latency

TLB latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

tlbi_stall_enabled

If true, tlb invalidation broadcast requests will block the requesting PE until all the other PEs flush their TLBs.

Type

bool

Default value

0x0

treat_PAC_as_NOP

Non-architecture parameter. Treat Pointer Authentication as NOP. When the parameter is true behaviour of FEAT_PAuth instructions changes as following PAC* and AUT* and XPAC* instructions are NOP. Brach, load, return with authentication instructions do not check and change signature bits and ignore FEAT_PAuth traps.

Type

bool

Default value

0x0

walk_cache_latency

Walk cache latency for TA (Timing Annotation), expressed in simulation ticks

Type

int

Default value

0x0

3.5.83 ARMSC000CT

ARMSC000CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-523: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMSC000CT

This model has the following Iris instances:

Table 3-524: ARMSC000CT Iris instances

InstanceName	ComponentName
ARMSC000CT	ARM_SC000
ARMSC000CT.acp_mapper	PVBusMapper
ARMSC000CT.ext_bus	PVBusLogger
ARMSC000CT.ext_bus.mapper	PVBusMapper
ARMSC000CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-525: ARMSC000CT MTI instances

InstanceName	ComponentName
ARMSC000CT	ARM_SC000
ARMSC000CT.acp_mapper	PVBusMapper
ARMSC000CT.ext_bus	PVBusLogger
ARMSC000CT.ext_bus.mapper	PVBusMapper
ARMSC000CT.l2_flusher	AsyncCacheFlushUnit

ARMSC000CT contains the following CADI targets:

- ARM_SC000

Differences between the model and the RTL

The model has the following limitations:

- It does not implement any security features.
- Only bit[0] of the Auxiliary Control Register is supported for read/write. No functionality is implemented.
- The Security Features Control Register read/write access is supported using SECKEY. No functionality is implemented.

Ports for ARMSC000CT

Table 3-526: Ports

Name	Protocol	Type	Description
ahbd	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.
currpri	Value	Master	Current execution priority.
edbgrq	Signal	Slave	External debug request.

Name	Protocol	Type	Description
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
intisr[32]	Signal	Slave	This signal array delivers signals to the NVIC.
intnmi	Signal	Slave	Configure non maskable interrupt.
io_port_in	PVBus	Slave	I/O port pair. See the documentation for the io_port_out port.
io_port_out	PVBus	Master	I/O port pair. Used if IOP is true. Transactions from io_port_out which do not "match" should be returned via io_port_in. For performance reasons, the I/O port interface is not modelled directly. Instead, a simple PVBus gasket is inserted at the point in the memory system where the I/O port would be. In hardware, a device would be attached to the port which would tell the CPU whether it would like to intercept each transaction, given its address. This can be modelled in a performant manner by connecting a PVBusMapper-based device to io_port_out which intercepts transactions of interest and passes other transactions back to the CPU via io_port_in. Your I/O port device model is also responsible for aborting transactions which would be aborted on hardware (e.g. exclusives) if necessary.
lockup	Signal	Master	Asserted when the processor is in lockup state.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbush_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
stcalib	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.

Parameters for ARMSC000CT

BIGENDINIT

Initialize processor to big endian mode

Type

bool

Default value

0x0

BKPT

Number of breakpoint unit comparators implemented

Type

int

Default value

0x4

DBG

Set whether debug extensions are implemented

Type

bool

Default value

0x1

IOP

Send all d-side transactions to the port, io_port_out. Transactions which do not match should be returned to the port, io_port_in

Type

bool

Default value

0x0

IRQDIS

IRQ line disable mask. Bit n of this 32-bit parameter disables IRQ[n]

Type

int

Default value

0x0

NUM_IRQ

Number of user interrupts

Type

int

Default value

0x20

NUM_MPU_REGION

Number of MPU regions

Type

int

Default value

0x0

SYST

Enable support for SysTick timer functionality

Type

bool

Default value

0x1

USER

Enable support for Unprivileged/Privileged Extension

Type

bool

Default value

0x1

VTOR

Include Vector Table Offset Register

Type

bool

Default value

0x1

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

WPT

Number of watchpoint unit comparators implemented

Type

int

Default value

0x2

cpi_div

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base

Virtual address of base of descending stack

Type

int

Default value

0x20800000

semihosting-stack_limit

Virtual address of stack limit

Type

int

Default value

0x20700000

3.5.84 ARMSC300CT

ARMSC300CT CPU component. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-527: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ARMSC300CT

This model has the following Iris instances:

Table 3-528: ARMSC300CT Iris instances

InstanceName	ComponentName
ARMSC300CT	ARM_SC300
ARMSC300CT.acp_mapper	PVBusMapper
ARMSC300CT.ext_bus	PVBusLogger
ARMSC300CT.ext_bus.mapper	PVBusMapper
ARMSC300CT.l2_flusher	AsyncCacheFlushUnit

This model has the following MTI trace components:

Table 3-529: ARMSC300CT MTI instances

InstanceName	ComponentName
ARMSC300CT	ARM_SC300
ARMSC300CT.acp_mapper	PVBusMapper
ARMSC300CT.ext_bus	PVBusLogger
ARMSC300CT.ext_bus.mapper	PVBusMapper
ARMSC300CT.l2_flusher	AsyncCacheFlushUnit

ARMSC300CT contains the following CADI targets:

- ARM_SC300

Differences between the model and the RTL

The model has the following limitations:

- It does not implement any security features.
- The Trash Register is implemented as RAZ/WI.
- Only bit[0] of the Auxiliary Control Register is supported for read/write. No functionality is implemented.
- The Security Features Control Register read/write access is supported using SECKEY. No functionality is implemented.

Implementation of ITM in M-class models

This model has a parameter that enables partial support for Instrumentation Trace Macrocell (ITM). In hardware or RTL, trace data from ITM is sent in packets to the trace block serially using a single pin or wire. In the model, if it is enabled, the ITM trace data is output using an MTI trace source called `ITM`. The `ITM` trace source has an `ITM_PACKET_TYPE` field. The following table shows which packet types the model supports:

Table 3-530: ITM_PACKET_TYPE field values that the model supports

Field value	Description	Supported by model
ITM_SYNC	Synchronization packet	Not supported.
ITM_P_OVERFLOW	Protocol: Overflow packet	Not supported.
ITM_P_LOCAL_TIMESTAMP	Protocol: Local timestamp packets	Not supported.
ITM_P_GLOBAL_TIMESTAMP	Protocol: Global timestamp packets	Not supported.
ITM_P_EXTEN	Protocol: Extension packet	Not supported.
ITM_S_INSTRUMENTATION	Source: Instrumentation packet	Supported.
ITM_S_DWT_EVENT_COUNTER	Hardware source: Event counter wrapping	Not supported.
ITM_S_DWT_EXCEPTION	Hardware source: Exception tracing	Supported.
ITM_S_DWT_PC_SAMPLING	Hardware source: PC sampling	Not supported.
ITM_S_DWT_DATA_PC_TRACE	Hardware source: DWT Data trace PC value	Supported.
ITM_S_DWT_DATA_ADDRESS_TRACE	Hardware source: DWT Data trace address value	Supported.
ITM_S_DWT_DATA_DATA_TRACE	Hardware source: DWT Data trace DATA value	Supported.

Ports for ARMSC300CT

Table 3-531: Ports

Name	Protocol	Type	Description
ahbd	PVBus	Slave	Debug AHB - core bus slave driven by the DAP.
auxfault	Value	Slave	This is wired to the Auxiliary Fault Status Register.
bigend	Signal	Slave	Configure big endian data format.
clk_in	ClockSignal	Slave	The clock signal connected to the clk_in port is used to determine the rate at which the core executes instructions.

Name	Protocol	Type	Description
currpri	Value	Master	Current execution priority.
edbgrq	Signal	Slave	External debug request.
event	Signal	Peer	This peer port of event input (and output) is for wakeup from WFE and corresponds to the RTL TXEV and RXEV signals.
intisr[240]	Signal	Slave	This signal array delivers signals to the NVIC.
intnmi	Signal	Slave	Configure non maskable interrupt.
lockup	Signal	Master	Asserted when the processor is in lockup state.
poreset	Signal	Slave	Raising this signal will do a power-on reset of the core.
pv_ppbus_m	PVBus	Master	The core will generate External Private Peripheral Bus requests on this port.
pvbus_m	PVBus	Master	The core will generate bus requests on this port.
sleepdeep	Signal	Master	Asserted when the processor is in deep sleep.
sleeping	Signal	Master	Asserted when the processor is in sleep.
stcalib	Value	Slave	This is the calibration value for the SysTick timer.
stclk	ClockSignal	Slave	This is the reference clock for the SysTick timer.
sysreset	Signal	Slave	Raising this signal will put the core into reset mode (but does not reset the debug logic).
sysresetreq	Signal	Master	Asserted to indicate that a reset is required.
ticks	InstructionCount	Master	Port allowing the number of instructions since startup to be read from the CPU.

Parameters for ARMSC300CT

BB_PRESENT

Enable bitbanding

Type

bool

Default value

0x1

BIGENDINIT

Initialize processor to big endian mode

Type

bool

Default value

0x0

DBGLVL

0: No debug; 1: Minimal debug; 2: Full debug without DWT address matching; 3: Full debug support with DWT data-comparators

Type

int

Default value

0x3

LVL_WIDTH

Number of bits of interrupt priority

Type

int

Default value

0x3

NUM_IRQ

Number of user interrupts

Type

int

Default value

0x10

NUM_MPU_REGION

Number of MPU regions

Type

int

Default value

0x8

TRACE_LVL

Level of instrumentation trace supported. 0: No trace, 1: Standard trace. ITM, TPIU and DWT triggers and counters present, 2: Full trace. ITM, TPIU, ETM and DWT triggers and counters present

Type

int

Default value

0x1

WIC

Include support for WIC-mode deep sleep

Type

bool

Default value

0x1

cpi_div

divider for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

cpi_mul

multiplier for calculating CPI (Cycles Per Instruction)

Type

int

Default value

0x1

master_id

Master ID presented in bus transactions

Type

int

Default value

0x0

min_sync_level

force minimum syncLevel (0=off=default, 1=syncState, 2=postInsnIO, 3=postInsnAll)

Type

int

Default value

0x0

reported_patch_level

Purely cosmetic patch level value to be displayed from the CPUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

reported_revision_number

Purely cosmetic revision number value to be displayed from the CPUID register. This value will not necessarily align with the model behaviour. The default value (-1) will see this parameter ignored.

Type

int

Default value

0xffffffffffffffff

semihosting-Thumb_SVC

T32 SVC number for semihosting

Type

int

Default value

0xab

semihosting-cmd_line

Command line available to semihosting SVC calls

Type

string

Default value**semihosting-cwd**

Base directory for semihosting file access.

Type

string

Default value**semihosting-enable**

Enable semihosting SVC traps. Applications that do not use semihosting must set this parameter to false.

Type

bool

Default value

0x1

semihosting-heap_base

Virtual address of heap base

Type

int

Default value

0x0

semihosting-heap_limit

Virtual address of top of heap

Type

int

Default value

0x20700000

semihosting-stack_base
Virtual address of base of descending stack

Type
int

Default value
0x20800000

semihosting-stack_limit
Virtual address of stack limit

Type
int

Default value
0x20700000

3.6 Media components

This section describes the Media components.

3.6.1 D71

ARM D71 Display Processor. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-532: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for D71

This model has the following Iris instances:

Table 3-533: D71 Iris instances

InstanceName	ComponentName
D71	D71
D71.apb_slave_adu	PVBusSlave
D71.apb_slave_dpu	PVBusSlave

This model has the following MTI trace components:

Table 3-534: D71 MTI instances

InstanceName	ComponentName
D71	D71
D71.apb_slave_adu	PVBusSlave
D71.apb_slave_dpu	PVBusSlave

D71 contains the following CADI targets:

- D71

About D71

The model has the following limitations:

- No support for trusted layers.
- No support for image enhancements.
- No coprocessor support for HDR processing.
- No QoS support.
- The following configuration parameters are not available:
 - CONFIG_MAX_LINE_SIZE
 - CONFIG_DISPLAY_TBU_EN. TBUs are integrated separately using the given ports.
 - CONFIG_AFBC_DMA_EN. The ADU is present. If it is not used, do not program it.

Ports for D71

Table 3-535: Ports

Name	Protocol	Type	Description
apb_pvbus_s_adu	PVBus	Slave	Slave port for register access.
apb_pvbus_s_dpu	PVBus	Slave	-
axi_pvbus_adu_m	PVBus	Master	Master AXI port for the AFBC unit
axi_pvbus_lpu_m[2]	PVBus	Master	Master AXI ports for pipelines
display[2]	LCD	Master	LCD ports for display outputs
irq0_gcu_out	Signal	Master	Shared interrupt owned by the GCU
irq1_adu_out	Signal	Master	Interrupt signal for the ADU block
pixelclock_in[2]	ClockSignal	Slave	Pixel clock inputs for the display outputs
pvbus_tbu_m[2]	PVBus	Master	Master ports for connection to TBU (SMMUv3)
pvbus_tbu_s[2]	PVBus	Slave	Slave ports for loopback from TBU (SMMUv3)
reset_signal	Signal	Slave	Reset signal.

Parameters for D71

adu_nprot_nsaid

Non-protected NSAID for ADU transactions

Type

int

Default value

0x0

adu_nprot_s2_sid

Stage 2 non-protected StreamID for ADU transactions

Type

int

Default value

0x2

adu_prot_nsa_id

Protected NSAID for ADU transactions

Type

int

Default value

0x1

adu_prot_s2_sid

Stage 2 protected StreamID for ADU transactions

Type

int

Default value

0x5

adu_rd_s1_sid

Stage 1 StreamID for ADU DMA read layer

Type

int

Default value

0xa

adu_wr_s1_sid

Stage 1 StreamID for ADU AES write-back layer

Type

int

Default value

0xb

display_split_en

Display split enabled or not

Type

int

Default value

0x0

force_frame_rate_0

If 0 PXL CLOCK0 is used, if >0 the model refreshes display output 0 at the rate per simulated second

Type

int

Default value

0x0

force_frame_rate_1

If 0 PXL CLOCK1 is used, if >0 the model refreshes display output 1 at the rate per simulated second

Type

int

Default value

0x0

lpu0_l0_s1_sid

Stage 1 StreamID for LPU0 read layer 0

Type

int

Default value

0x0

lpu0_l1_s1_sid

Stage 1 StreamID for LPU0 read layer 1

Type

int

Default value

0x1

lpu0_l2_s1_sid

Stage 1 StreamID for LPU0 read layer 2

Type

int

Default value

0x2

lpu0_l3_s1_sid

Stage 1 StreamID for LPU0 read layer 3

Type

int

Default value

0x3

lpu0_nprot_nsaaid

Non-protected NSAID for LPU0 transactions

Type

int

Default value

0x0

lpu0_nprot_s2_sid

Stage 2 non-protected StreamID for LPU0 transactions

Type

int

Default value

0x0

lpu0_prot_nsaaid

Protected NSAID for LPU0 transactions

Type

int

Default value

0x1

lpu0_prot_s2_sid

Stage 2 protected StreamID for LPU0 transactions

Type

int

Default value

0x3

lpu0_wr_s1_sid

Stage 1 StreamID for LPU0 write-back layer

Type

int

Default value

0x8

lpu1_10_s1_sid

Stage 1 StreamID for LPU1 read layer 0

Type

int

Default value

0x4

lpu1_11_s1_sid

Stage 1 StreamID for LPU1 read layer 1

Type

int

Default value

0x5

lpu1_12_s1_sid

Stage 1 StreamID for LPU1 read layer 2

Type

int

Default value

0x6

lpu1_13_s1_sid

Stage 1 StreamID for LPU1 read layer 3

Type

int

Default value

0x7

lpu1_nprot_nsa_id

Non-protected NSAID for LPU1 transactions

Type

int

Default value

0x0

lpul_nprot_s2_sid

Stage 2 non-protected StreamID for LPU1 transactions

Type

int

Default value

0x1

lpul_prot_nsaid

Protected NSAID for LPU1 transactions

Type

int

Default value

0x1

lpul_prot_s2_sid

Stage 2 protected StreamID for LPU1 transactions

Type

int

Default value

0x4

lpul_wr_s1_sid

Stage 1 StreamID for LPU1 write-back layer

Type

int

Default value

0x9

num_rich_layers

Number of Rich layers in each Layer Processing Unit

Type

int

Default value

0x2

3.6.2 DP500

ARM DP500 Display Processor. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-536: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DP500

This model has the following Iris instances:

Table 3-537: DP500 Iris instances

InstanceName	ComponentName
DP500	DP500
DP500.busmaster	PVBusMaster
DP500.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-538: DP500 MTI instances

InstanceName	ComponentName
DP500	DP500
DP500.busmaster	PVBusMaster
DP500.busslave	PVBusSlave

DP500 contains the following CADI targets:

- DP500

About DP500

This model has basic support for the display and scaling engines. Connect it to a visualization component to view LCD output. This is the single display configuration of DP500. For the dual display configuration, use DP500x2.

It passes tests as part of a booting Android kernel and running under the control of the official DP500 drivers.

It provides the following functionality:

- All RGB and YUV format parsing.
- Color adjustment in *Display Engine* (DE).
- Nearest neighbor scaling.
- All layers.
- Alpha blending.
- Memory writeback.
- Inverse gamma adjustment.

- Basic layer (overlay) and register security semantics.

The model has the following limitations:

- No support for polyphase scaling algorithm. Falls back to nearest neighbor when configured to do so.
- No support for 3D or interlaced video.
- No support for image enhancing functionality.
- No colorspace conversion support.
- No support for two plane YUV memory writeback.

Ports for DP500

Table 3-539: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Master clock input, typically 24MHz, to drive pixel clock timing.
display	LCD	Master	Connection to visualization component.
intr	Signal	Master	Interrupt signal.
intr_se	Signal	Master	Interrupt signal from scaling engine.
pvbus_m	PVBus	Master	Bus for processor 0.
pvbus_s	PVBus	Slave	Slave port for register access.
reset_signal	Signal	Slave	Reset signal.

Parameters for DP500

disable_snooping_dma

Disable DMA snooping

Type

bool

Default value

0x0

dual_mode

Whether 2 display mode is supported

Type

bool

Default value

0x0

force_frame_rate

If 0 - the input clock is used as PXLCKLOCK, if >0 then the model ensures the screen display is refreshed n times per simulated second

Type

int

Default value

0x0

3.6.3 DP500x2

ARM DP500 Display Processor x2. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-540: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DP500x2

This model has the following Iris instances:

Table 3-541: DP500x2 Iris instances

InstanceName	ComponentName
DP500x2	DP500x2
DP500x2.display_core_0	DP500
DP500x2.display_core_0.busmaster	PVBusMaster
DP500x2.display_core_0.busslave	PVBusSlave
DP500x2.display_core_1	DP500
DP500x2.display_core_1.busmaster	PVBusMaster
DP500x2.display_core_1.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-542: DP500x2 MTI instances

InstanceName	ComponentName
DP500x2.display_core_0	DP500
DP500x2.display_core_0.busmaster	PVBusMaster
DP500x2.display_core_0.busslave	PVBusSlave
DP500x2.display_core_1	DP500
DP500x2.display_core_1.busmaster	PVBusMaster
DP500x2.display_core_1.busslave	PVBusSlave

DP500x2 contains the following CADI targets:

- DP500
- DP500x2

About DP500x2

This component is a model of the dual display configuration of the DP500 Display Processor, with basic support for the display and scaling engines. Connect it to a visualization component to view LCD output.

The model provides the following functionality:

- All RGB and YUV format parsing.
- Color adjustment in *Display Engine* (DE).
- Nearest neighbor scaling.
- All layers.
- Alpha blending.
- Memory writeback.
- Inverse gamma adjustment.
- Basic layer (overlay) and register security semantics.

The model has the following limitations:

- No support for polyphase scaling algorithm. Falls back to nearest neighbor when configured to do so.
- No support for 3D or interlaced video.
- No support for image enhancing functionality.
- No colorspace conversion support.
- No support for two plane YUV memory writeback.

Ports for DP500x2

Table 3-543: Ports

Name	Protocol	Type	Description
dc_de_interrupt[2]	Signal	Master	Interrupt signalling from display engines.
dc_pvbus_m[2]	PVBus	Master	Bus for processor 0 and 1.
dc_se_interrupt[2]	Signal	Master	Interrupt signalling from scaling engines.
display[2]	LCD	Master	Connection to visualization component.
dp0_clk_in	ClockSignal	Slave	Clock signal for DP0.
dp1_clk_in	ClockSignal	Slave	Clock signal for DP1.
pvbus_s	PVBus	Slave	Slave port for register access.
reset_signal	Signal	Slave	Reset signal.

Parameters for DP500x2

disable_snooping_dma

Disable DMA snooping

Type

bool

Default value

0x0

force_frame_rate

If 0 - the input clock is used as PXLCLK, if >0 then the model ensures the screen display is refreshed n times per simulated second

Type

int

Default value

0x0

3.6.4 DP550

ARM DP550 Display Processor. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-544: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DP550

This model has the following Iris instances:

Table 3-545: DP550 Iris instances

InstanceName	ComponentName
DP550	DP550
DP550.busmaster	PVBusMaster
DP550.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-546: DP550 MTI instances

InstanceName	ComponentName
DP550	DP550

InstanceName	ComponentName
DP550.busmaster	PVBusMaster
DP550.busslave	PVBusSlave

DP550 contains the following CADI targets:

- DP550

About DP550

This component is a model of the DP550 Display Processor. Connect it to a visualization component to view LCD output. This is the single display configuration of DP550. For the dual display configuration, use DP550x2.

The model provides the following functionality:

- All RGB and YUV format parsing.
- Color adjustment in Display Engine (DE).
- Nearest neighbor scaling.
- All layers.
- Alpha blending.
- Memory writeback.
- Inverse gamma adjustment.
- Basic layer (overlay) and register security semantics.

The model has the following limitations:

- No support for polyphase scaling algorithm. Falls back to nearest neighbor when configured to do so.
- No support for 3D or interlaced video.
- No support for image enhancing functionality.
- No colorspace conversion support.
- No support for two plane YUV memory writeback.

Ports for DP550

Table 3-547: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Master clock input, typically 24MHz, to drive pixel clock timing.
display	LCD	Master	Connection to visualization component.
intr	Signal	Master	Interrupt signal.
intr_se	Signal	Master	Interrupt signal from scaling engine.
pvbus_m	PVBus	Master	Bus for processor 0.
pvbus_s	PVBus	Slave	Slave port for register access.
reset_signal	Signal	Slave	Reset signal.

Parameters for DP550

disable_snooping_dma

Disable DMA snooping

Type

bool

Default value

0x0

dual_mode

Whether 2 display mode is supported

Type

bool

Default value

0x0

force_frame_rate

If 0 - the input clock is used as PXLCLK, if >0 then the model ensures the screen display is refreshed n times per simulated second

Type

int

Default value

0x0

set_nprot_nsaids

Configurable NPROT_NSAID for non-secure NSAID values

Type

int

Default value

0xc00ba98

set_nprot_streamids

Configurable NPROT_STREAMID for non-secure STREAMID values

Type

int

Default value

0xc00ba98

set_prot_nsaids

Configurable PROT_NSAID for secure NSAID values

Type

int

Default value
0x4003210

set_prot_streamid
Configurable PROT_STREAMID for secure STREAMID values

Type
int

Default value
0x4003210

3.6.5 DP550x2

ARM DP550 Display Processor x2. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-548: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DP550x2

This model has the following Iris instances:

Table 3-549: DP550x2 Iris instances

InstanceName	ComponentName
DP550x2	DP550x2
DP550x2.display_core_0	DP550
DP550x2.display_core_0.busmaster	PVBusMaster
DP550x2.display_core_0.busslave	PVBusSlave
DP550x2.display_core_1	DP550
DP550x2.display_core_1.busmaster	PVBusMaster
DP550x2.display_core_1.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-550: DP550x2 MTI instances

InstanceName	ComponentName
DP550x2.display_core_0	DP550
DP550x2.display_core_0.busmaster	PVBusMaster
DP550x2.display_core_0.busslave	PVBusSlave

InstanceName	ComponentName
DP550x2.display_core_1	DP550
DP550x2.display_core_1.busmaster	PVBusMaster
DP550x2.display_core_1.busslave	PVBusSlave

DP550x2 contains the following CADI targets:

- DP550
- DP550x2

About DP550x2

This component is a model of the dual display configuration of the DP550 Display Processor. Connect it to a visualization component to view LCD output.

The model provides the following functionality:

- All RGB and YUV format parsing.
- Color adjustment in *Display Engine* (DE).
- Nearest neighbor scaling.
- All layers.
- Alpha blending.
- Memory writeback.
- Inverse gamma adjustment.
- Basic layer (overlay) and register security semantics.

The model has the following limitations:

- No support for polyphase scaling algorithm. Falls back to nearest neighbor when configured to do so.
- No support for 3D or interlaced video.
- No support for image enhancing functionality.
- No colorspace conversion support.
- No support for two plane YUV memory writeback.

Ports for DP550x2

Table 3-551: Ports

Name	Protocol	Type	Description
dc_de_interrupt[2]	Signal	Master	Interrupt signalling from display engines.
dc_pvbus_m[2]	PVBus	Master	Bus for processor 0 and 1.
dc_se_interrupt[2]	Signal	Master	Interrupt signalling from scaling engines.
display[2]	LCD	Master	Connection to visualization component.
dp0_clk_in	ClockSignal	Slave	Clock signal for DPO.

Name	Protocol	Type	Description
dp1_clk_in	ClockSignal	Slave	Clock signal for DP1.
pvbus_s	PVBus	Slave	Slave port for register access.
reset_signal	Signal	Slave	Reset signal.

Parameters for DP550x2

disable_snooping_dma

Disable DMA snooping

Type

bool

Default value

0x0

force_frame_rate

If 0 - the input clock is used as PXLCLK, if >0 then the model ensures the screen display is refreshed n times per simulated second

Type

int

Default value

0x0

set_primary_nprot_nsaids

Configurable NPROT_NSAID for non-secure NSAID values(primary)

Type

int

Default value

0xc00ba98

set_primary_nprot_streamids

Configurable NPROT_STREAMID for non-secure STREAMID values(primary)

Type

int

Default value

0xc00ba98

set_primary_prot_nsaids

Configurable PROT_NSAID for secure NSAID values(primary)

Type

int

Default value

0x4003210

set_primary_prot_streamid

Configurable PROT_STREAMID for secure STREAMID values(primary)

Type

int

Default value

0x4003210

set_secondary_nprot_nsaids

Configurable NPROT_NSAID for non-secure NSAID values(secondary)

Type

int

Default value

0xc00ba98

set_secondary_nprot_streamid

Configurable NPROT_STREAMID for non-secure STREAMID values(secondary)

Type

int

Default value

0xc00ba98

set_secondary_prot_nsaids

Configurable PROT_NSAID for secure NSAID values(secondary)

Type

int

Default value

0x4003210

set_secondary_prot_streamid

Configurable PROT_STREAMID for secure STREAMID values(secondary)

Type

int

Default value

0x4003210

3.6.6 DP650

ARM DP650 Display Processor. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-552: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DP650

This model has the following Iris instances:

Table 3-553: DP650 Iris instances

InstanceName	ComponentName
DP650	DP650
DP650.busmaster	PVBusMaster
DP650.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-554: DP650 MTI instances

InstanceName	ComponentName
DP650	DP650
DP650.busmaster	PVBusMaster
DP650.busslave	PVBusSlave

DP650 contains the following CADI targets:

- DP650

About DP650

This component is a model of the DP650 Display Processor. Connect it to a visualization component to view LCD output. This is the single display configuration of DP650. For the dual display configuration, use DP650x2.

The model has the following limitations:

- No support for the polyphase scaling algorithm, it always uses nearest neighbor.
- No support for 3D or interlaced video.
- No support for image enhancing functionality.
- No YUV 2-plane support for memory writeback.
- No color space conversion support.

Ports for DP650

Table 3-555: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Master clock input, typically 24MHz, to drive pixel clock timing.

Name	Protocol	Type	Description
display	LCD	Master	Connection to visualization component.
intr	Signal	Master	Interrupt signal.
intr_se	Signal	Master	Interrupt signal from scaling engine.
pvbuse_m	PVBus	Master	Bus for processor 0.
pvbuse_s	PVBus	Slave	Slave port for register access.
reset_signal	Signal	Slave	Reset signal.

Parameters for DP650

disable_snooping_dma

Disable DMA snooping

Type

bool

Default value

0x0

dual_mode

Whether 2 display mode is supported

Type

bool

Default value

0x0

force_frame_rate

If 0 - the input clock is used as PXLCLK, if >0 then the model ensures the screen display is refreshed n times per simulated second

Type

int

Default value

0x0

set_nprot_nsaid

Configurable NPROT_NSAID for non-secure NSAID values

Type

int

Default value

0xc00ba98

set_nprot_streamid

Configurable NPROT_STREAMID for non-secure STREAMID values

Type

int

Default value

0xc00ba98

set_prot_nsaidth

Configurable PROT_NSAID for secure NSAID values

Type

int

Default value

0x4003210

set_prot_streamidth

Configurable PROT_STREAMID for secure STREAMID values

Type

int

Default value

0x4003210

3.6.7 DP650x2

ARM DP650 Display Processor x2. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-556: IP revisions support

Revision	Quality level
rOp0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DP650x2

This model has the following Iris instances:

Table 3-557: DP650x2 Iris instances

InstanceName	ComponentName
DP650x2	DP650x2
DP650x2.display_core_0	DP650
DP650x2.display_core_0.busmaster	PVBusMaster
DP650x2.display_core_0.busslave	PVBusSlave
DP650x2.display_core_1	DP650
DP650x2.display_core_1.busmaster	PVBusMaster

InstanceName	ComponentName
DP650x2.display_core_1.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-558: DP650x2 MTI instances

InstanceName	ComponentName
DP650x2.display_core_0	DP650
DP650x2.display_core_0.busmaster	PVBusMaster
DP650x2.display_core_0.busslave	PVBusSlave
DP650x2.display_core_1	DP650
DP650x2.display_core_1.busmaster	PVBusMaster
DP650x2.display_core_1.busslave	PVBusSlave

DP650x2 contains the following CADI targets:

- DP650
- DP650x2

About DP650x2

This component is a model of the dual display configuration of the DP650 Display Processor. To view LCD output, connect it to a visualization component.

The model has the following limitations:

- No support for the polyphase scaling algorithm, it always uses nearest neighbor.
- No support for 3D or interlaced video.
- No support for image enhancing functionality.
- No YUV 2-plane support for memory writeback.
- No color space conversion support.

Ports for DP650x2

Table 3-559: Ports

Name	Protocol	Type	Description
dc_de_interrupt[2]	Signal	Master	Interrupt signalling from display engines.
dc_pvbus_m[2]	PVBus	Master	Bus for processor 0 and 1.
dc_se_interrupt[2]	Signal	Master	Interrupt signalling from scaling engines.
display[2]	LCD	Master	Connection to visualization component.
dp0_clk_in	ClockSignal	Slave	Clock signal for DP0.
dp1_clk_in	ClockSignal	Slave	Clock signal for DP1.
pvbus_s	PVBus	Slave	Slave port for register access.
reset_signal	Signal	Slave	Reset signal.

Parameters for DP650x2

disable_snooping_dma

Disable DMA snooping

Type

bool

Default value

0x0

force_frame_rate

If 0 - the input clock is used as PXLCLK, if >0 then the model ensures the screen display is refreshed n times per simulated second

Type

int

Default value

0x0

set_primary_nprot_nsaids

Configurable NPROT_NSAID for non-secure NSAID values(primary)

Type

int

Default value

0xc00ba98

set_primary_nprot_streamids

Configurable NPROT_STREAMID for non-secure STREAMID values(primary)

Type

int

Default value

0xc00ba98

set_primary_prot_nsaids

Configurable PROT_NSAID for secure NSAID values(primary)

Type

int

Default value

0x4003210

set_primary_prot_streamids

Configurable PROT_STREAMID for secure STREAMID values(primary)

Type

int

Default value

0x4003210

set_secondary_nprot_nsaids

Configurable NPROT_NSAID for non-secure NSAID values(secondary)

Type

int

Default value

0xc00ba98

set_secondary_nprot_streamids

Configurable NPROT_STREAMID for non-secure STREAMID values(secondary)

Type

int

Default value

0xc00ba98

set_secondary_prot_nsaids

Configurable PROT_NSAID for secure NSAID values(secondary)

Type

int

Default value

0x4003210

set_secondary_prot_streamids

Configurable PROT_STREAMID for secure STREAMID values(secondary)

Type

int

Default value

0x4003210

3.6.8 Mali_C5x_streaming_sink

Arm® Mali™-C55 ISP streaming output capture example component. This component tests ISP's streaming output functionality and provides an example of how to develop components consuming frames from the Mali-C55 ISP streaming output. This implementation saves all the ISP output frames to host disk in the FRM file format which is defined as part of the ISP model specification. All the implementation code is placed in this LISA+ file. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-560: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_C5x_streaming_sink

This model has the following Iris instances:

Table 3-561: Mali_C5x_streaming_sink Iris instances

InstanceName	ComponentName
Mali_C5x_streaming_sink	Mali_C5x_streaming_sink
Mali_C5x_streaming_sink.bus_slave0	PVBusSlave
Mali_C5x_streaming_sink.bus_slave1	PVBusSlave
Mali_C5x_streaming_sink.bus_slave2	PVBusSlave

This model has the following MTI trace components:

Table 3-562: Mali_C5x_streaming_sink MTI instances

InstanceName	ComponentName
Mali_C5x_streaming_sink.bus_slave0	PVBusSlave
Mali_C5x_streaming_sink.bus_slave1	PVBusSlave
Mali_C5x_streaming_sink.bus_slave2	PVBusSlave

Mali_C5x_streaming_sink contains the following CADI targets:

- [Mali_C5x_streaming_sink](#)

Ports for Mali_C5x_streaming_sink

Table 3-563: Ports

Name	Protocol	Type	Description
data_s[3]	PVBus	Slave	Pixels consumer ports; receive pixels as 16-bit values (and optional debug metadata).
hsync_s	Signal	Slave	Horizontal sync port; defines input image's line begin and end for all three data ports.
vsync_s	Signal	Slave	Vertical sync port; defines the input frame's begin and end for all three data ports.

Parameters for Mali_C5x_streaming_sink

do_capture

Saving captured frames on/off

Type

bool

Default value

0x1

fn_prefix

Saved filenames prefix

Type

string

Default value

isp_out_

3.6.9 Mali_C7x_streaming_sink

Arm® Mali™-C71/C78 ISP streaming output capture example component. This component tests ISP's streaming output functionality and provides an example of how to develop components consuming frames from the Mali-C7x ISP streaming output. This implementation saves all the ISP output frames to host disk in the FRM file format which is defined as a part of the ISP model specification. All the implementation code is placed in this LISA+ file. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-564: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_C7x_streaming_sink

This model has the following Iris instances:

Table 3-565: Mali_C7x_streaming_sink Iris instances

InstanceName	ComponentName
Mali_C7x_streaming_sink	Mali_C7x_streaming_sink
Mali_C7x_streaming_sink.bus_slave0	PVBusSlave
Mali_C7x_streaming_sink.bus_slave1	PVBusSlave
Mali_C7x_streaming_sink.bus_slave2	PVBusSlave

This model has the following MTI trace components:

Table 3-566: Mali_C7x_streaming_sink MTI instances

InstanceName	ComponentName
Mali_C7x_streaming_sink.bus_slave0	PVBusSlave
Mali_C7x_streaming_sink.bus_slave1	PVBusSlave
Mali_C7x_streaming_sink.bus_slave2	PVBusSlave

Mali_C7x_streaming_sink contains the following CADI targets:

- Mali_C7x_streaming_sink

Ports for Mali_C7x_streaming_sink

Table 3-567: Ports

Name	Protocol	Type	Description
data_s[3]	PVBus	Slave	Pixels consumer ports; receive pixels as 16-bit values (and optional debug metadata).
hsync_s[3]	Signal	Slave	Horizontal sync ports; defines input image's line begin and end for the corresponding data ports.
vsync_s[3]	Signal	Slave	Vertical sync ports; define input frame's begin and end for the corresponding data port.

Parameters for Mali_C7x_streaming_sink

do_capture

Saving captured frames on/off

Type

bool

Default value

0x1

fn_prefix

Saved filenames prefix

Type

string

Default value

isp_out_

3.6.10 Mali_C55

Arm® Mali™ C55 ISP. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-568: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_C55

This model has the following Iris instances:

Table 3-569: Mali_C55 Iris instances

InstanceName	ComponentName
Mali_C55	Mali_C55
Mali_C55.apb_slave	PVBusSlave

InstanceName	ComponentName
Mali_C55.bus_slave0	PVBusSlave
Mali_C55.bus_slave1	PVBusSlave
Mali_C55.bus_slave2	PVBusSlave

This model has the following MTI trace components:

Table 3-570: Mali_C55 MTI instances

InstanceName	ComponentName
Mali_C55.apb_slave	PVBusSlave
Mali_C55.bus_slave0	PVBusSlave
Mali_C55.bus_slave1	PVBusSlave
Mali_C55.bus_slave2	PVBusSlave

Mali_C55 contains the following CADI targets:

- Mali_C55

Mali_C55 parameters

hw_config

Enable or disable optional features. These features are controlled by virtual registers, any number of which can be configured by this parameter, separated by spaces.

Type: str. Default value: \$FRSCALER_FITTED=1 \$PONG_CONFIG_FITTED=1.

The hw_config parameter supports the following virtual registers and assignments:

\$CNR_FITTED

0

CNR and its square/sqrt and the relative config space are statically removed from the ISP. The CNR and its square/sqrt config address space are not writeable and reading it back returns 0.

1

CNR and its square/sqrt are present in the design.

Default value: 1.

\$COMPRESSION_FITTED

0

Temper compression is statically removed from the ISP.

1

Temper compression is present in the design.

Default value: 1.

\$DSPPIPE_FITTED**0**

Downscale pipe and the relative config space are statically removed from the ISP. The downscale pipe config address space is not writeable and reading it back returns 0.

1

Downscale pipe is present in the design.

Default value: 1.

\$FRSCALER_FITTED**0**

Full resolution pipe scaler and the relative config space are statically removed from the ISP. The full resolution pipe scaler config address space is not writeable and reading it back returns 0.

1

Full resolution pipe scaler is present in the design.

Default value: 0.

\$IRIDIX_GTM_FITTED**0**

Iridix global tone mapping is removed from the ISP.

1

Iridix global tone mapping is present in the design.

Default value: 0.

\$IRIDIX_LTM_FITTED**0**

Iridix local tone mapping is removed from the ISP.

1

Iridix local tone mapping is present in the design.

Default value: 1.

\$PONG_CONFIG_FITTED**0**

Pong configuration space, with the exception of pong statistics memories, is statically removed from the ISP. The address space relative to the pong configuration space is not writeable and reading back the address space returns 0.

1

Pong configuration space is present in the design.

Default value: 1.

\$SCALER_COEF_SETS

Number of scaler coefficient sets available. This value is expressed in decimal.

Default value: 8.

\$SINTER_FITTED

0

Sinter and the relative config space are statically removed from the ISP. The sinter config address space is not writeable and reading it back returns 0.

1

Sinter is present in the design.

Default value: 1.

\$SINTER_LITE

0

Full sinter version used.

1

Sinter lite version used.

Default value: 0.

\$TEMPER_FITTED

0

Temper and the relative config space are statically removed from the ISP. The temper config address space is not writeable and reading it back returns 0.

1

Temper is present in the design.

Default value: 1.

\$WDR_FITTED

0

WDR Frame Stitch, offset and gain and the relative config registers are statically removed from the ISP.

1

WDR Frame Stitch, offset and gain are present in the design.

Default value: 1.

Limitations

The Mali_C55 ISP model has the following limitations:

- Streaming input and output interfaces are supported but the protocol differs slightly from the real hardware. See `Mali_cxx_streaming_camera.lisa` and `Mali_c5x_streaming_sink.lisa` for details on how to use them.

- Video monitor output and DMA monitor interface are not supported.
- Error conditions (IRQ bits 2, 3, 19, 20, 22) are not supported.
- It does not expose any targets for CADI or Iris-enabled debuggers to connect to, so debug access to its registers through CADI or Iris is not supported.

Ports for Mali_C55

Table 3-571: Ports

Name	Protocol	Type	Description
ds_data_out_m[3]	PVBus	Master	-
ds_hsync_out_m	Signal	Master	-
ds_uv_valid_out_m	Signal	Master	-
ds_vsync_out_m	Signal	Master	-
fr_data_out_m[3]	PVBus	Master	Metadata + pixels, 16 bit, (RGB or YUV)
fr_hsync_out_m	Signal	Master	Horizontal sync / blanking (set at the beginning of a line; cleared at the end)
fr_uv_valid_out_m	Signal	Master	UV valid (set if both U and V pixels are valid)
fr_vsync_out_m	Signal	Master	Vertical sync / blanking (set at the beginning of a frame; cleared at the end)
irq	Signal	Master	Shared interrupt
pvbus_m	PVBus	Master	Master AXI port for RAM access
pvbus_s	PVBus	Slave	Slave port for register access
reset_s	Signal	Slave	Reset signal
stream_data_in_s[3]	PVBus	Slave	Metadata + pixels, 20 bit
stream_hsync_in_s	Signal	Slave	Horizontal sync / blanking (set at the beginning of a line; cleared at the end)
stream_vsync_in_s	Signal	Slave	Vertical sync / blanking (set at the beginning of a frame; cleared at the end)

Parameters for Mali_C55

hw_config

Setting on/off optional features. See 'Virtual registers' section in 'C Model Integration Guide'

Type

string

Default value

\$FRSCALER_FITTED=1 \$PONG_CONFIG_FITTED=1

3.6.11 Mali_C71

Arm® Mali™ C71 ISP. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-572: IP revisions support

Revision	Quality level
r1p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Ports added:

- `fault`

Iris and MTI instances for Mali_C71

This model has the following Iris instances:

Table 3-573: Mali_C71 Iris instances

InstanceName	ComponentName
Mali_C71	Mali_C71
Mali_C71.apb_slave	PVBusSlave
Mali_C71.bus_slave0	PVBusSlave
Mali_C71.bus_slave1	PVBusSlave
Mali_C71.bus_slave2	PVBusSlave
Mali_C71.bus_slave3	PVBusSlave

This model has the following MTI trace components:

Table 3-574: Mali_C71 MTI instances

InstanceName	ComponentName
Mali_C71.apb_slave	PVBusSlave
Mali_C71.bus_slave0	PVBusSlave
Mali_C71.bus_slave1	PVBusSlave
Mali_C71.bus_slave2	PVBusSlave
Mali_C71.bus_slave3	PVBusSlave

Mali_C71 contains the following CADI targets:

- Mali_C71

Model limitations

The Mali_C71 ISP model has the following limitations:

- Streaming input and output interfaces are supported but the protocol differs slightly from the real hardware.
- Fault Interface is not supported.
- Testing interrupt connectivity through the Interrupt Trigger register at offset 0x118DC is not supported.

- Double interrupt signal is not supported.
- It does not expose any targets for CADI or Iris-enabled debuggers to connect to, so debug access to its registers through CADI or Iris is not supported.

Ports for Mali_C71

Table 3-575: Ports

Name	Protocol	Type	Description
fault	Signal	Master	Fault output interface
irq[4]	Signal	Master	Shared interrupts
pvbus_m	PVBus	Master	Master AXI port for RAM access
pvbus_s	PVBus	Slave	Slave port for register access
reset_s	Signal	Slave	Reset signal.
stream_data_in_s[4]	PVBus	Slave	Metadata + pixels
stream_data_out_m[3]	PVBus	Master	Metadata + pixels
stream_hsync_in_s[4]	Signal	Slave	Horizontal sync / blanking (set at the beginning of a line; cleared at the end)
stream_hsync_out_m[3]	Signal	Master	Horizontal sync / blanking (set at the beginning of a line; cleared at the end)
stream_tag_in_s[4]	Value	Slave	Source stream tag Can change between lines (while hsync is Clear) if several frames are multiplexed.
stream_tag_out_m[3]	Value	Master	Source stream tag (changes while vsync is Clear)
stream_vsync_in_s[4]	Signal	Slave	Vertical sync / blanking (set at the beginning of a frame; cleared at the end)
stream_vsync_out_m[3]	Signal	Master	Vertical sync / blanking (set at the beginning of a frame; cleared at the end)

3.6.12 Mali_C78

Arm® Mali™ C78 ISP. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-576: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Table 3-577: IP revision changes

From	To
r0p0	r1p0

Ports added:

- fault

Iris and MTI instances for Mali_C78

This model has the following Iris instances:

Table 3-578: Mali_C78 Iris instances

InstanceName	ComponentName
Mali_C78	Mali_C78
Mali_C78.apb_slave	PVBusSlave
Mali_C78.bus_slave0	PVBusSlave
Mali_C78.bus_slave1	PVBusSlave
Mali_C78.bus_slave2	PVBusSlave
Mali_C78.bus_slave3	PVBusSlave

This model has the following MTI trace components:

Table 3-579: Mali_C78 MTI instances

InstanceName	ComponentName
Mali_C78.apb_slave	PVBusSlave
Mali_C78.bus_slave0	PVBusSlave
Mali_C78.bus_slave1	PVBusSlave
Mali_C78.bus_slave2	PVBusSlave
Mali_C78.bus_slave3	PVBusSlave

Mali_C78 contains the following CADI targets:

- Mali_C78

Model limitations

The Mali_C78 ISP model has the following limitations:

- Streaming input and output interfaces are supported but the protocol differs slightly from the real hardware.
- Fault Interface is not supported.
- Testing interrupt connectivity through the Interrupt Trigger register at offset 0x118DC is not supported.
- Double interrupt signal is not supported.
- It does not expose any targets for CADI or Iris-enabled debuggers to connect to, so debug access to its registers through CADI or Iris is not supported.

Ports for Mali_C78

Table 3-580: Ports

Name	Protocol	Type	Description
fault	Signal	Master	Fault output interface
irq[4]	Signal	Master	Shared interrupts

Name	Protocol	Type	Description
pvbuse_m	PVBus	Master	Master AXI port for RAM access
pvbuse_s	PVBus	Slave	Slave port for register access
reset_s	Signal	Slave	Reset signal.
stream_data_in_s[4]	PVBus	Slave	Metadata + pixels
stream_data_out_m[3]	PVBus	Master	Metadata + pixels
stream_hsync_in_s[4]	Signal	Slave	Horizontal sync / blanking (set at the beginning of a line; cleared at the end)
stream_hsync_out_m[3]	Signal	Master	Horizontal sync / blanking (set at the beginning of a line; cleared at the end)
stream_tag_in_s[4]	Value	Slave	Source stream tag Can change between lines (while hsync is Clear) if several frames are multiplexed.
stream_tag_out_m[3]	Value	Master	Source stream tag (changes while vsync is Clear)
stream_vsync_in_s[4]	Signal	Slave	Vertical sync / blanking (set at the beginning of a frame; cleared at the end)
stream_vsync_out_m[3]	Signal	Master	Vertical sync / blanking (set at the beginning of a frame; cleared at the end)

3.6.13 Mali_Cxx_streaming_camera

Simple streaming camera to connect to Mali ISP (C55, C71, or C78) streaming inputs. This component tests ISP's streaming input functionality and provides an example of how to develop camera-like components streaming frames to a Mali-Cxx ISP. All the implementation code is placed in this LISA+ file. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-581: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_Cxx_streaming_camera

This model has the following Iris instances:

Table 3-582: Mali_Cxx_streaming_camera Iris instances

InstanceName	ComponentName
Mali_Cxx_streaming_camera	Mali_Cxx_streaming_camera
Mali_Cxx_streaming_camera.bus_master_data	PVBusMaster
Mali_Cxx_streaming_camera.bus_slave_config	PVBusSlave

This model has the following MTI trace components:

Table 3-583: Mali_Cxx_streaming_camera MTI instances

InstanceName	ComponentName
Mali_Cxx_streaming_camera.bus_master_data	PVBusMaster
Mali_Cxx_streaming_camera.bus_slave_config	PVBusSlave

Mali_Cxx_streaming_camera contains the following CADI targets:

- Mali_Cxx_streaming_camera

Ports for Mali_Cxx_streaming_camera

Table 3-584: Ports

Name	Protocol	Type	Description
config_s	PVBus	Slave	Access to the camera config register(s) (8 bit): 0x00 - output mode: 0 - no output, 1 - stream one frame and auto-reset to 0
data_m	PVBus	Master	Output frames port; sends pixels as 32-bit values (and debug metadata).
hsync_m	Signal	Master	Horizontal sync; set at the beginning of a line, cleared at the end.
vsync_m	Signal	Master	Vertical sync; set at the beginning of a frame, cleared at the end.

Parameters for Mali_Cxx_streaming_camera

image_file

A file containing one or more frames to stream in the ISP model's FRM format

Type

string

Default value

3.6.14 Mali_G51

ARM Mali-G51 GPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-585: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_G51

This model has the following Iris instances:

Table 3-586: Mali_G51 Iris instances

InstanceName	ComponentName
Mali_G51	Mali_G51

InstanceName	ComponentName
Mali_G51.busmaster	PVBusMaster
Mali_G51.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-587: Mali_G51 MTI instances

InstanceName	ComponentName
Mali_G51	Mali_G51
Mali_G51.busmaster	PVBusMaster
Mali_G51.busslave	PVBusSlave

Mali_G51 contains the following CADI targets:

- Mali_G51

About Mali_G51

The model has the following limitations:

- It does not support Arm®v8-style page tables.
- It does not execute GPU shader programs.
- It does not validate all register values or job descriptors.

This model outputs values for all Mali™ PMU hardware counters. These values are not meaningful. Their purpose is to enable you to use the model early in the development process to check that your system is correctly configured to support performance analysis tools, for instance Streamline.

The counter is a 32-bit bitfield, with the following bit assignments:

[31:28]

Identifies which Mali GPU instance generated this value, typically zero for a system with a single GPU.

[27:24]

Identifies which hardware block is the source of the counter. It can have one of the following values:

- | | |
|----|------------------------|
| 0 | Job manager. |
| 1 | Tiler. |
| 2 | L2Cache/Memory system. |
| 3+ | Shader core. |

[23:16]

The counter number within the block.

[15:0]

Sawtooth counter that counts from 0 to `0xffff` and then resets to 0. This value ensures that consecutive captures show different values and allows you to check that counter values are changing over time.



- These counter values might change in future.
- The model always generates counter values, even if the GPU is idle.
- Streamline does not display the precise values that the model outputs because it needs to correct them for sampling frequency and other profiling effects. However, their size relative to each other is correct. For example, counters from the job manager are always smaller than those from the memory system.

Ports for Mali_G51**Table 3-588: Ports**

Name	Protocol	Type	Description
gpu_reset	Signal	Slave	Resets the GPU at the input Set. The available inputs are Set and Clear.
irq_gpu	Signal	Master	The interrupt signal generated from the GPU.
irq_job	Signal	Master	The interrupt signal generated from the Job Manager on the GPU.
irq_mmu	Signal	Master	The interrupt signal generated from the MMU on the GPU.
prot_mode_m0	Signal	Master	Indicates the current state of the GPU. When it outputs: Set: The GPU is in protected mode. Clear: The GPU is in normal mode. Typically, connect this signal to the LabellerForGPUProtMode component. This enables the labeller to add a Non-Secure Access ID to the outgoing transaction on the labeller's pvbus_m port.
pvbus_m	PVBus	Master	The interface for the GPU to access external memory.
pvbus_s	PVBus	Slave	The interface for the CPU to access the GPU registers.

Parameters for Mali_G51**revision**

Revision of the RTL that the model represents. Valid values: r0p0

Type

string

Default value

r0p0

3.6.15 Mali_G71

ARM Mali-G71 GPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-589: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_G71

This model has the following Iris instances:

Table 3-590: Mali_G71 Iris instances

InstanceName	ComponentName
Mali_G71	Mali_G71
Mali_G71.busmaster	PVBusMaster
Mali_G71.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-591: Mali_G71 MTI instances

InstanceName	ComponentName
Mali_G71	Mali_G71
Mali_G71.busmaster	PVBusMaster
Mali_G71.busslave	PVBusSlave

Mali_G71 contains the following CADI targets:

- Mali_G71

About Mali_G71

The model has the following limitations:

- It does not support Arm®v8-style page tables.
- It does not execute GPU shader programs.
- It does not validate all register values or job descriptors.

This model outputs values for all Mali™ PMU hardware counters. These values are not meaningful. Their purpose is to enable you to use the model early in the development process to check that your system is correctly configured to support performance analysis tools, for instance Streamline.

The counter is a 32-bit bitfield, with the following bit assignments:

[31:28]

Identifies which Mali™ GPU instance generated this value, typically zero for a system with a single GPU.

[27:24]

Identifies which hardware block is the source of the counter. It can have one of the following values:

- 0Job manager.
- 1Tiler.
- 2L2Cache/Memory system.
- 3+Shader core.

[23:16]
The counter number within the block.

[15:0]
Sawtooth counter that counts from 0 to 0xffff and then resets to 0. This value ensures that consecutive captures show different values and allows you to check that counter values are changing over time.



- These counter values might change in future.
- The model always generates counter values, even if the GPU is idle.
- Streamline does not display the precise values that the model outputs because it needs to correct them for sampling frequency and other profiling effects. However, their size relative to each other is correct. For example, counters from the job manager are always smaller than those from the memory system.

Ports for Mali_G71

Table 3-592: Ports

Name	Protocol	Type	Description
gpu_reset	Signal	Slave	Resets the GPU at the input Set. The available inputs are Set and Clear.
irq_gpu	Signal	Master	The interrupt signal generated from the GPU.
irq_job	Signal	Master	The interrupt signal generated from the Job Manager on the GPU.
irq_mmu	Signal	Master	The interrupt signal generated from the MMU on the GPU.
prot_mode_m0	Signal	Master	Indicates the current state of the GPU. When it outputs: Set: The GPU is in protected mode. Clear: The GPU is in normal mode. Typically, connect this signal to the LabellerForGPUProtMode component. This enables the labeller to add a Non-Secure Access ID to the outgoing transaction on the labeller's pvbus_m port.
pvbus_m	PVBus	Master	The interface for the GPU to access external memory.
pvbus_s	PVBus	Slave	The interface for the CPU to access the GPU registers.

Parameters for Mali_G71

revision
Revision of the RTL that the model represents. Valid values: rOp0

Type
string

Default value
rOp0

3.6.16 Mali_G72

ARM Mali-G72 GPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-593: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_G72

This model has the following Iris instances:

Table 3-594: Mali_G72 Iris instances

InstanceName	ComponentName
Mali_G72	Mali_G72
Mali_G72.busmaster	PVBusMaster
Mali_G72.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-595: Mali_G72 MTI instances

InstanceName	ComponentName
Mali_G72	Mali_G72
Mali_G72.busmaster	PVBusMaster
Mali_G72.busslave	PVBusSlave

Mali_G72 contains the following CADI targets:

- Mali_G72

About Mali_G72

The model has the following limitations:

- It does not support Arm®v8-style page tables.
- It does not execute GPU shader programs.
- It does not validate all register values or job descriptors.

This model outputs values for all Mali™ PMU hardware counters. These values are not meaningful. Their purpose is to enable you to use the model early in the development process to check that your system is correctly configured to support performance analysis tools, for instance Streamline.

The counter is a 32-bit bitfield, with the following bit assignments:

[31:28]
Identifies which Mali GPU instance generated this value, typically zero for a system with a single GPU.

[27:24]
Identifies which hardware block is the source of the counter. It can have one of the following values:

- 0Job manager.
- 1Tiler.
- 2L2Cache/Memory system.
- 3+Shader core.

[23:16]
The counter number within the block.

[15:0]
Sawtooth counter that counts from 0 to 0xffff and then resets to 0. This value ensures that consecutive captures show different values and allows you to check that counter values are changing over time.



- These counter values might change in future.
- The model always generates counter values, even if the GPU is idle.
- Streamline does not display the precise values that the model outputs because it needs to correct them for sampling frequency and other profiling effects. However, their size relative to each other is correct. For example, counters from the job manager are always smaller than those from the memory system.

Ports for Mali_G72

Table 3-596: Ports

Name	Protocol	Type	Description
gpu_reset	Signal	Slave	Resets the GPU at the input Set. The available inputs are Set and Clear.
irq_gpu	Signal	Master	The interrupt signal generated from the GPU.
irq_job	Signal	Master	The interrupt signal generated from the Job Manager on the GPU.
irq_mmu	Signal	Master	The interrupt signal generated from the MMU on the GPU.
prot_mode_m0	Signal	Master	Indicates the current state of the GPU. When it outputs: Set: The GPU is in protected mode. Clear: The GPU is in normal mode. Typically, connect this signal to the LabellerForGPUProtMode component. This enables the labeller to add a Non-Secure Access ID to the outgoing transaction on the labeller's pvbus_m port.
pvbus_m	PVBus	Master	The interface for the GPU to access external memory.
pvbus_s	PVBus	Slave	The interface for the CPU to access the GPU registers.

Parameters for Mali_G72

revision
Revision of the RTL that the model represents. Valid values: rOp0

Type
string

Default value
rOp0

3.6.17 Mali_G76

ARM Mali-G76 GPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-597: IP revisions support

Revision	Quality level
rOp0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_G76

This model has the following Iris instances:

Table 3-598: Mali_G76 Iris instances

InstanceName	ComponentName
Mali_G76	Mali_G76
Mali_G76.busmaster	PVBusMaster
Mali_G76.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-599: Mali_G76 MTI instances

InstanceName	ComponentName
Mali_G76	Mali_G76
Mali_G76.busmaster	PVBusMaster
Mali_G76.busslave	PVBusSlave

Mali_G76 contains the following CADI targets:

- Mali_G76

About Mali_G76

The model has the following limitations:

- It does not support Arm®v8-style page tables.
- It does not execute GPU shader programs.
- It does not validate all register values or job descriptors.

This model outputs values for all Mali™ PMU hardware counters. These values are not meaningful. Their purpose is to enable you to use the model early in the development process to check that your system is correctly configured to support performance analysis tools, for instance Streamline.

The counter is a 32-bit bitfield, with the following bit assignments:

[31:28]

Identifies which Mali GPU instance generated this value, typically zero for a system with a single GPU.

[27:24]

Identifies which hardware block is the source of the counter. It can have one of the following values:

0	Job manager.
1	Tiler.
2	L2Cache/Memory system.
3+	Shader core.

[23:16]

The counter number within the block.

[15:0]

Sawtooth counter that counts from 0 to 0xffff and then resets to 0. This value ensures that consecutive captures show different values and allows you to check that counter values are changing over time.



Note

- These counter values might change in future versions.
- The model always generates counter values, even if the GPU is idle.
- Streamline does not display the precise values that the model outputs because it needs to correct them for sampling frequency and other profiling effects. However, their size relative to each other is correct. For example, counters from the job manager are always smaller than those from the memory system.

Ports for Mali_G76

Table 3-600: Ports

Name	Protocol	Type	Description
gpu_reset	Signal	Slave	Resets the GPU at the input Set. The available inputs are Set and Clear.
irq_gpu	Signal	Master	The interrupt signal generated from the GPU.
irq_job	Signal	Master	The interrupt signal generated from the Job Manager on the GPU.
irq_mmu	Signal	Master	The interrupt signal generated from the MMU on the GPU.
prot_mode_m0	Signal	Master	Indicates the current state of the GPU. When it outputs: Set: The GPU is in protected mode. Clear: The GPU is in normal mode. Typically, connect this signal to the LabellerForGPUProtMode component. This enables the labeller to add a Non-Secure Access ID to the outgoing transaction on the labeller's pvbush_m port.
pvbush_m	PVBus	Master	The interface for the GPU to access external memory.

Name	Protocol	Type	Description
pvbuss_s	PVBus	Slave	The interface for the CPU to access the GPU registers.

Parameters for Mali_G76

revision

Revision of the RTL that the model represents. Valid values: rOp0

Type

string

Default value

rOp0

3.6.18 Mali_G78AE

Arm® Mali™-G78AE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-601: IP revisions support

Revision	Quality level
rOp0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_G78AE

This model has the following Iris instances:

Table 3-602: Mali_G78AE Iris instances

InstanceName	ComponentName
Mali_G78AE	Mali_G78AE
Mali_G78AE.AccessControl	PVBusMapper

This model has the following MTI trace components:

Table 3-603: Mali_G78AE MTI instances

InstanceName	ComponentName
Mali_G78AE	Mali_G78AE
Mali_G78AE.AccessControl	PVBusMapper

Mali_G78AE contains the following CADI targets:

- Mali_G78AE

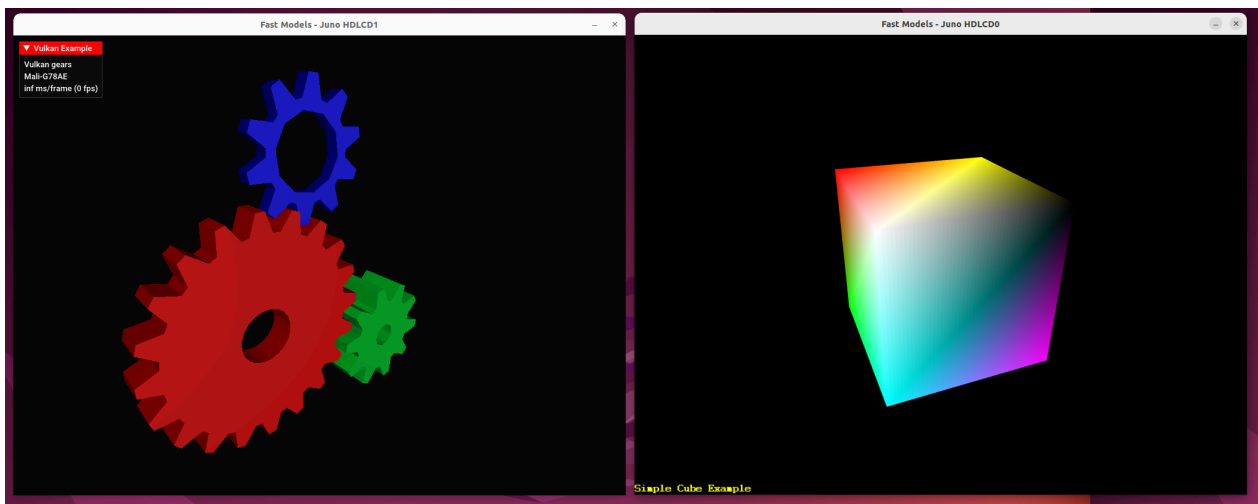
About Mali_G78AE

The Mali_G78AE component models the Arm® Mali™ G78AE, which implements the Valhall architecture, and is the first Mali GPU designed specifically to target automotive use cases.

The model implements the Partition Manager, enabling up to 4 independent partitions running workloads at the same time, while being accessed by up to 16 virtual machines.

The Mali_G78AE model is functional, capable of executing GPU shader programs and producing graphical or compute outputs. It requires a clock input running at around 500MHz to keep the CPU and GPU performance aligned, and to avoid software timeouts during slow-running GPU operations.

Figure 3-2: Mali_G78AE example frames



With this model, and the implemented functionality of the Partition Manager, you can:

- Simulate the entire graphics software stack, including the user space and kernel space driver components of the Arm Mali DDK, and an application that uses a graphics API.
- Verify the integration of the Mali G78AE GPU into the rest of the platform. A complex use case example of this is a system running multiple kernels in Virtual Machines under a Hypervisor, all submitting workloads to the GPU at the same time.

To configure the reference Mali driver for use with the model, we recommend you make some adjustments to the timing parameters of the Mali driver. This is because of timing differences between the real hardware and the Fast Model. Which parameters work best depend on your system, but if you are using the reference Arm implementation of the arbiter from the DDK, one possibility is:

```
insmod mali_kbase.ko gpu_req_timeout=1000
insmod mali_arbiter.ko request_timeout=200 yield_timeout=30
```

Test applications

The tests have been carried out using the r40p0 release of the Mali DDK. The following applications have been tested and confirmed to work:

- A selection of the Mali DDK integration tests in `product/build-<wsi>/install/bin/...`
 - mali_gles_integration_suite
 - mali_cl_simple_example
- A selection of lightweight Vulkan examples hosted on [GitHub](#):
 - gears
 - computeparticles, at small particle count
 - texture3d
- A selection of ComputeLibrary examples hosted on [GitHub](#):
 - graph_lenet
 - graph_mobilenet_v2

All of these applications have been successfully tested in a non-virtualised system, on Debian Buster running Linux 4.19.

A subset of these examples have been tested in the following virtualised systems:

- Xen Hypervisor 4.14.1-pre
- Privileged Debian Buster running Linux kernel 4.19, controlling the configuration of partitions, and running the Mali DDK reference arbiter implementation
- Two unprivileged virtual machines, both running Debian Buster, Linux 4.19, both running various applications listed in this section at the same time

Limitations

The Mali_G78AE model has the following limitations:

- It does not support debug access to registers through CADI or Iris-enabled debuggers.
- It is not supported on Windows hosts.
- It is a functional model and does not simulate performance differences for partitions of different sizes.
- It does not implement the protection *CHK signals.
- It does not implement Parity/DCLS/CRC faults and fault fingerprints.
- It does not implement different error response modes configurable through the SYSTEM page.

Ports for Mali_G78AE

Table 3-604: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	-
gpu_reset	Signal	Slave	Reset signals

Name	Protocol	Type	Description
gpu_reset_recovery	Signal	Slave	-
irq_deferred_error	Signal	Master	-
irq_groups[4]	Signal	Master	-
irq_partitions[4]	Signal	Master	Partition Manager irqs
irq_uncorrected_error	Signal	Master	-
irq_windows[16]	Signal	Master	-
pvbus_m	PVBus	Master	Output to board from GPU
pvbus_s[3]	PVBus	Slave	Slave bus ports (AXI-A through C).
sys_assign_enable	Value	Slave	System configuration access control for different ports

Parameters for Mali_G78AE

labeller_encoding_spec

Specification of how the StreamID is encoded into transaction attributes

Type

string

Default value

MasterID[31:0]=StreamID[31:0]

3.6.19 Mali_G710

Arm® Mali™ G710 GPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-605: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_G710

This model has the following Iris instances:

Table 3-606: Mali_G710 Iris instances

InstanceName	ComponentName
Mali_G710	Mali_G710
Mali_G710.busmaster	PVBusMaster
Mali_G710.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-607: Mali_G710 MTI instances

InstanceName	ComponentName
Mali_G710	Mali_G710
Mali_G710.busmaster	PVBusMaster
Mali_G710.busslave	PVBusSlave

Mali_G710 contains the following CADI targets:

- Mali_G710

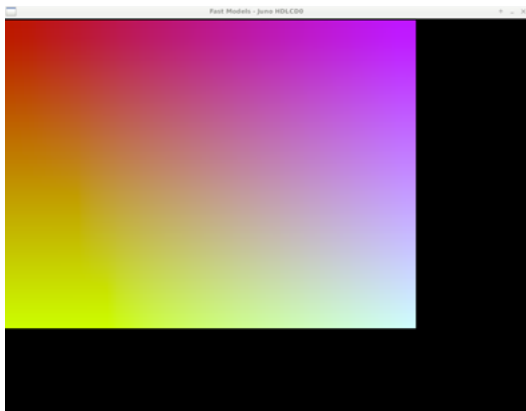
About Mali_G710 and Mali_G715

The Mali_G710 and Mali_G715 components model the Arm® Mali™ G710 and G715 GPUs respectively, which implement the Valhall architecture. They are among the first Mali GPUs to implement a Command Stream Frontend (CSF), which is a combination of hardware and firmware that helps to offload work from the Mali driver that is running on the CPU.

With the Mali G710 and G715 Fast Models, you can simulate:

- User space and kernel space driver components of the Arm Mali Driver Development kit (Mali DDK), when built to target Mali G710 or G715 under Android and Linux graphics stacks. See later in this topic for details of supported window systems.
- Mali G710 or G715 firmware binaries.

These are functional models that do not attempt to execute GPU shader programs. They are sufficient to simulate a Mali driver, but rather than deliver fully rendered images to memory, they write a randomly-generated color gradient pattern to the framebuffer, similar to the following:

Figure 3-3: Example frame from Mali_G710

The changing frames can be used to prove that the graphics stack is correctly configured to allow the Mali GPU to write to a user-visible framebuffer.

Configuring the driver for use with the model

To account for timing differences between real hardware and the Fast Model, it is necessary to modify the following runtime settings in the Mali driver:

Table 3-608: Mali driver runtime settings

Parameter name	Type	Recommended value	Description
csf_firmware_boot_timeout_ms	Kernel module parameter	10000	Overrides the minimum timeout value for loading firmware into the model.
reset_timeout	Sysfs parameter	1000000	Overrides the minimum timeout when waiting for operations on the GPU to finish.
fw_timeout	Sysfs parameter	1000000	Overrides the minimum timeout when waiting for operations on the GPU to finish.

The following commands are a snippet from an Android RC file that is an example of how a filesystem can be configured to correctly load the Mali driver when running on a Fast Models platform:

```
insmod /vendor/lib/modules/mali_kbase.ko csf_firmware_boot_timeout_ms=100000
wait /dev/mali0
chmod 0666 /dev/mali0
wait /sys/class/misc/mali0/device/reset_timeout
write /sys/class/misc/mali0/device/reset_timeout 1000000
write /sys/class/misc/mali0/device/fw_timeout 1000000
```

Limitations

- The model does not support debug access to registers through CADI or Iris-enabled debuggers.
- The timing data observed from the model does not correspond in any way to real hardware.
- The model does not expose any MTI targets.
- The model is not supported on Windows hosts.

Ports for Mali_G710

Table 3-609: Ports

Name	Protocol	Type	Description
gpu_reset	Signal	Slave	Resets the GPU at the input Set. The available inputs are Set and Clear.
irq_gpu	Signal	Master	The interrupt signal generated from the GPU.
irq_job	Signal	Master	The interrupt signal generated from the Job Manager on the GPU.
irq_mmu	Signal	Master	The interrupt signal generated from the MMU on the GPU.
prot_mode_m0	Signal	Master	Indicates the current state of the GPU. When it outputs: Set: The GPU is in protected mode. Clear: The GPU is in normal mode. Typically, connect this signal to the LabellerForGPUProtMode component. This enables the labeller to add a Non-Secure Access ID to the outgoing transaction on the labeller's pvbus_m port.
pvbus_m	PVBus	Master	The interface for the GPU to access external memory.
pvbus_s	PVBus	Slave	The interface for the CPU to access the GPU registers.

Parameters for Mali_G710

altcmdline

Alternate command line for the GPU model. If used, mode is ignored.

Type

string

Default value

altmodel

Path to an alternative GPU model library

Type

string

Default value

mode

GPU Mode. Inputs supported: [fast, turbo_fallback, turbo]

Type

string

Default value

fast

3.6.20 Mali_G715

Arm® Mali™ G715 GPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-610: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_G715

This model has the following Iris instances:

Table 3-611: Mali_G715 Iris instances

InstanceName	ComponentName
Mali_G715	Mali_G715
Mali_G715.busmaster	PVBusMaster
Mali_G715.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-612: Mali_G715 MTI instances

InstanceName	ComponentName
Mali_G715	Mali_G715
Mali_G715.busmaster	PVBusMaster
Mali_G715.busslave	PVBusSlave

Mali_G715 contains the following CADI targets:

- Mali_G715

About Mali_G710 and Mali_G715

The Mali_G710 and Mali_G715 components model the Arm® Mali™ G710 and G715 GPUs respectively, which implement the Valhall architecture. They are among the first Mali GPUs to implement a Command Stream Frontend (CSF), which is a combination of hardware and firmware that helps to offload work from the Mali driver that is running on the CPU.

With the Mali G710 and G715 Fast Models, you can simulate:

- User space and kernel space driver components of the Arm Mali Driver Development kit (Mali DDK), when built to target Mali G710 or G715 under Android and Linux graphics stacks. See later in this topic for details of supported window systems.
- Mali G710 or G715 firmware binaries.

These are functional models that do not attempt to execute GPU shader programs. They are sufficient to simulate a Mali driver, but rather than deliver fully rendered images to memory, they write a randomly-generated color gradient pattern to the framebuffer, similar to the following:

Figure 3-4: Example frame from Mali_G710

The changing frames can be used to prove that the graphics stack is correctly configured to allow the Mali GPU to write to a user-visible framebuffer.

Configuring the driver for use with the model

To account for timing differences between real hardware and the Fast Model, it is necessary to modify the following runtime settings in the Mali driver:

Table 3-613: Mali driver runtime settings

Parameter name	Type	Recommended value	Description
csf_firmware_boot_timeout_ms	Kernel module parameter	10000	Overrides the minimum timeout value for loading firmware into the model.
reset_timeout	Sysfs parameter	1000000	Overrides the minimum timeout when waiting for operations on the GPU to finish.
fw_timeout	Sysfs parameter	1000000	Overrides the minimum timeout when waiting for operations on the GPU to finish.

The following commands are a snippet from an Android RC file that is an example of how a filesystem can be configured to correctly load the Mali driver when running on a Fast Models platform:

```
insmod /vendor/lib/modules/mali_kbase.ko csf_firmware_boot_timeout_ms=100000
wait /dev/mali0
chmod 0666 /dev/mali0
wait /sys/class/misc/mali0/device/reset_timeout
write /sys/class/misc/mali0/device/reset_timeout 1000000
write /sys/class/misc/mali0/device/fw_timeout 1000000
```

Limitations

- The model does not support debug access to registers through CADI or Iris-enabled debuggers.
- The timing data observed from the model does not correspond in any way to real hardware.
- The model does not expose any MTI targets.
- The model is not supported on Windows hosts.

Ports for Mali_G715

Table 3-614: Ports

Name	Protocol	Type	Description
gpu_reset	Signal	Slave	Resets the GPU at the input Set. The available inputs are Set and Clear.
irq_gpu	Signal	Master	The interrupt signal generated from the GPU.
irq_job	Signal	Master	The interrupt signal generated from the Job Manager on the GPU.
irq_mmu	Signal	Master	The interrupt signal generated from the MMU on the GPU.
prot_mode_m0	Signal	Master	Indicates the current state of the GPU. When it outputs: Set: The GPU is in protected mode. Clear: The GPU is in normal mode. Typically, connect this signal to the LabellerForGPUProtMode component. This enables the labeller to add a Non-Secure Access ID to the outgoing transaction on the labeller's pvbus_m port.
pvbus_m	PVBus	Master	The interface for the GPU to access external memory.
pvbus_s	PVBus	Slave	The interface for the CPU to access the GPU registers.

Parameters for Mali_G715

altcmdline

Alternate command line for the GPU model. If used, mode is ignored.

Type

string

Default value

altmodel

Path to an alternative GPU model library

Type

string

Default value

mode

GPU Mode. Inputs supported: [fast, turbo_fallback, turbo]

Type

string

Default value

fast

3.6.21 Mali_G720

Arm® Mali™ G720 GPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-615: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_G720

This model has the following Iris instances:

Table 3-616: Mali_G720 Iris instances

InstanceName	ComponentName
Mali_G720	Mali_G720
Mali_G720.busmaster	PVBusMaster
Mali_G720.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-617: Mali_G720 MTI instances

InstanceName	ComponentName
Mali_G720	Mali_G720
Mali_G720.busmaster	PVBusMaster
Mali_G720.busslave	PVBusSlave

Mali_G720 contains the following CADI targets:

- Mali_G720

About Mali_G720

This model is the first of a new generation of fully-functional GPU models that generate correct output without the assistance of Generic Graphics Accelerator (GGA).

It supports x86_64 and AArch64 hosts running a supported version of Linux, as listed in [Requirements for Fast Models](#) in the *Fast Models User Guide*. It does not support Windows hosts.

\$ (PVLIB_HOME) / LISA / Mali_G720.lisa requires a clock input running at around 500MHz to keep CPU and GPU performance aligned. This avoids software timeouts during slow-running GPU operations.

From a Fast Models perspective, the differences between Arm® Mali™-G720, Arm Immortalis™-G720, and Mali-G620 are minor. The Fast Model does not model the caches and the core count is transparent to the user. Ray tracing is not currently modeled.

Limitations

- The model does not support debug access to registers through CADI or Iris-enabled debuggers.
- The timing data observed from the model does not correspond in any way to real hardware.
- The model is not supported on Windows hosts.

Ports for Mali_G720

Table 3-618: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	-
gpu_reset	Signal	Slave	Resets the GPU at the input Set. The available inputs are Set and Clear.
irq_gpu	Signal	Master	The interrupt signal generated from the GPU.
irq_job	Signal	Master	The interrupt signal generated from the Job Manager on the GPU.
irq_mmu	Signal	Master	The interrupt signal generated from the MMU on the GPU.
prot_mode_m0	Signal	Master	Indicates the current state of the GPU. When it outputs: Set: The GPU is in protected mode. Clear: The GPU is in normal mode. Typically, connect this signal to the LabellerForGPUProtMode component. This enables the labeller to add a Non-Secure Access ID to the outgoing transaction on the labeller's pvbus_m port.
pvbus_m	PVBus	Master	The interface for the GPU to access external memory.
pvbus_s	PVBus	Slave	The interface for the CPU to access the GPU registers.

Parameters for Mali_G720

- altcmdline**

Alternate command line for the GPU model. If used, mode is ignored.

Type

string

Default value
- altmodel**

Path to an alternative GPU model library

Type

string

Default value
- mode**

GPU Mode. Inputs supported: [fast, turbo_fallback, turbo]

Type

string

Default value

fast
- turbo_threads**

Number of threads used

Type

int

Default value

0x0

3.6.22 Mali_G725

Arm® Mali™ G725 GPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-619: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_G725

This model has the following Iris instances:

Table 3-620: Mali_G725 Iris instances

InstanceName	ComponentName
Mali_G725	Mali_G725
Mali_G725.busmaster	PVBusMaster
Mali_G725.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-621: Mali_G725 MTI instances

InstanceName	ComponentName
Mali_G725	Mali_G725
Mali_G725.busmaster	PVBusMaster
Mali_G725.busslave	PVBusSlave

Mali_G725 contains the following CADI targets:

- Mali_G725

About Mali_G725

This component is a model of the Arm® Mali™-G725 GPU, fully capable of executing shaders and producing graphical or compute outputs.

It supports x86_64 and AArch64 hosts running a supported version of Linux, as listed in [Requirements for Fast Models](#) in the *Fast Models User Guide*. It does not support Windows hosts.

`$(PVLIB_HOME)/LISA/Mali_G725.lisa` requires a clock input running at around 500MHz to keep CPU and GPU performance aligned. This avoids software timeouts during slow-running GPU operations.

From a Fast Models perspective, the differences between Mali-G725, Arm Immortalis™-G925, and Mali-G625 are minor. The Fast Model does not model the caches and the core count is transparent to the user. Ray tracing is not currently modeled.

Limitations

- The model does not support debug access to registers through CADI or Iris-enabled debuggers.
- The timing data observed from the model does not correspond in any way to real hardware.
- The model is not supported on Windows hosts.

Ports for Mali_G725

Table 3-622: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	-
gpu_reset	Signal	Slave	Resets the GPU at the input Set. The available inputs are Set and Clear.
irq_gpu	Signal	Master	The interrupt signal generated from the GPU.

Name	Protocol	Type	Description
irq_job	Signal	Master	The interrupt signal generated from the Job Manager on the GPU.
irq_mmu	Signal	Master	The interrupt signal generated from the MMU on the GPU.
prot_mode_m0	Signal	Master	Indicates the current state of the GPU. When it outputs: Set: The GPU is in protected mode. Clear: The GPU is in normal mode. Typically, connect this signal to the LabellerForGPUProtMode component. This enables the labeller to add a Non-Secure Access ID to the outgoing transaction on the labeller's pvbus_m port.
pvbus_m	PVBus	Master	The interface for the GPU to access external memory.
pvbus_s	PVBus	Slave	The interface for the CPU to access the GPU registers.

Parameters for Mali_G725

altcmdline

Alternate command line for the GPU model. If used, mode is ignored.

Type

string

Default value

altmodel

Path to an alternative GPU model library

Type

string

Default value

mode

GPU Mode. Inputs supported: [fast, turbo_fallback, turbo]

Type

string

Default value

fast

turbo_threads

Number of threads used

Type

int

Default value

0x0

3.6.23 Mali_T624

ARM Mali-T624 GPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-623: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Mali_T624

This model has the following Iris instances:

Table 3-624: Mali_T624 Iris instances

InstanceName	ComponentName
Mali_T624	Mali_T624
Mali_T624.busmaster	PVBusMaster
Mali_T624.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-625: Mali_T624 MTI instances

InstanceName	ComponentName
Mali_T624	Mali_T624
Mali_T624.busmaster	PVBusMaster
Mali_T624.busslave	PVBusSlave

Mali_T624 contains the following CADI targets:

- Mali_T624

Ports for Mali_T624

Table 3-626: Ports

Name	Protocol	Type	Description
gpu_reset	Signal	Slave	Resets the GPU at the input Set. The available inputs are Set and Clear.
irq_gpu	Signal	Master	The interrupt signal generated from the GPU.
irq_job	Signal	Master	The interrupt signal generated from the Job Manager on the GPU.
irq_mmu	Signal	Master	The interrupt signal generated from the MMU on the GPU.
prot_mode_m0	Signal	Master	Indicates the current state of the GPU. When it outputs: Set: The GPU is in protected mode. Clear: The GPU is in normal mode. Typically, connect this signal to the LabellerForGPUProtMode component. This enables the labeller to add a Non-Secure Access ID to the outgoing transaction on the labeller's pvbus_m port.
pvbus_m	PVBus	Master	The interface for the GPU to access external memory.

Name	Protocol	Type	Description
pvbuss_s	PVBus	Slave	The interface for the CPU to access the GPU registers.

Parameters for Mali_T624

revision

Revision of the RTL that the model represents. Valid values: rOp0

Type

string

Default value

rOp0

3.6.24 V61

Arm® Mali™-V61 Video Processor. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-627: IP revisions support

Revision	Quality level
rOp1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for V61

This model has the following Iris instances:

Table 3-628: V61 Iris instances

InstanceName	ComponentName
V61	V61
V61.BusModifier.LSID0	PVBusMapper
V61.BusModifier.LSID1	PVBusMapper
V61.BusModifier.LSID2	PVBusMapper
V61.BusModifier.LSID3	PVBusMapper
V61.apb_slave[0]	PVBusSlave

This model has the following MTI trace components:

Table 3-629: V61 MTI instances

InstanceName	ComponentName
V61	V61
V61.BusModifier.LSID0	PVBusMapper
V61.BusModifier.LSID1	PVBusMapper

InstanceName	ComponentName
V61.BusModifier.LSID2	PVBusMapper
V61.BusModifier.LSID3	PVBusMapper
V61.apb_slave[0]	PVBusSlave

V61 contains the following CADI targets:

- V61

About V61

The model requires an external OpenMAX (OMX) IL implementation for codec functionality. By default, V61 looks for `ffomaxil.dll` on Windows or `libffomaxil.so` on Linux in the model binary's directory or in the Fast Models installation. The default library path can be overridden using the `omx-library-path` parameter.

FFomaxIL is an OMX IL implementation provided by Arm in the TPIP package for convenience. Refer to the TPIP package for more details on FFomaxIL.

When querying the OMX core, V61 searches for the following roles in the list of OpenMAX components:

H.264 decode

"video_decoder.avc"

JPEG decode

"video_decoder.mjpeg"

MPEG2 decode

"video_decoder.mpeg2"

MPEG4 decode

"video_decoder.mpeg4"

VC1 decode

"video_decoder.vc1"

VP8 decode

"video_decoder.vp8"

VP8 encode

"video_encoder.vp8"



Note

To build example platforms containing V61, you must either install the TPIP package, or remove the dependency on `FFmpeg` and `libvpx` from the platform's `sgproj` file, by removing the line containing `v5xx.sgrepo`.

The model has the following limitations:

- No support for HEVC, VP9 and RealVideo decoders.

- No support for 10-bit video output.
- No support for RGB or AFBC input for encoding.
- No profiling support.
- No QoS support.
- Power/Test modes are modeled only as register state changes.

Ports for V61

Table 3-630: Ports

Name	Protocol	Type	Description
apb_s	PVBus	Slave	APB Slave port for register access.
axi_m	PVBus	Master	AXI master bus for memory accesses.
clk	ClockSignal	Slave	Master clock, typically 300MHz.
irq	Signal	Master	IRQ signal to host CPU.
reset	Signal	Slave	Reset signal.

Parameters for V61

AXI-data-width
AXI data width, logarithmic byte notation (3->64bit, 4->128 bit)
Type
int
Default value
0x4

enable-frame-rate-limiting
enable output rate control via the CLK port
Type
bool
Default value
0x0

fuse-disable-AFBC
disable AFBC support by fuse
Type
bool
Default value
0x0

fuse-disable-HEVC
disable HEVC support by fuse

Type

bool

Default value

0x0

`fuse-disable-Real`

disable RealVideo support by fuse

Type

bool

Default value

0x0

`fuse-disable-VPX`

disable VP8 support by fuse

Type

bool

Default value

0x0

`ncores`

Number of cores in the component

Type

int

Default value

0x1

`omx-library-path`

path to a user-provided OMX library; leave blank to use FFomxIL

Type

string

Default value**`supports-10bit`**

component supports 10-bit content decoding

Type

bool

Default value

0x1

`supports-64byte-ref-bursts`

component supports 64-byte bursts for reference pixel data

Type

bool

Default value

0x1

supports-encoding

component supports encoding

Type

bool

Default value

0x1

3.6.25 V550

Arm® Mali™-V550 Video Processor. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-631: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for V550

This model has the following Iris instances:

Table 3-632: V550 Iris instances

InstanceName	ComponentName
V550	V550
V550.BusModifier.LSID0	PVBusMapper
V550.BusModifier.LSID1	PVBusMapper
V550.BusModifier.LSID2	PVBusMapper
V550.BusModifier.LSID3	PVBusMapper
V550.apb_slave[0]	PVBusSlave

This model has the following MTI trace components:

Table 3-633: V550 MTI instances

InstanceName	ComponentName
V550	V550
V550.BusModifier.LSID0	PVBusMapper

InstanceName	ComponentName
V550.BusModifier.LSID1	PVBusMapper
V550.BusModifier.LSID2	PVBusMapper
V550.BusModifier.LSID3	PVBusMapper
V550.apb_slave[0]	PVBusSlave

V550 contains the following CADI targets:

- V550

About V550

The model requires an external OpenMAX (OMX) IL implementation for codec functionality. By default, V550 looks for `ffomaxil.dll` on Windows or `libffomaxil.so` on Linux in the model binary's directory or in the Fast Models installation. The default library path can be overridden using the parameter `omx-library-path`.

FFomaxIL is an OMX IL implementation that Arm provides for convenience in the Third-Party IP (TPIP) add-on package. For more details on FFomaxIL, refer to the TPIP package, which is available from [Product Download Hub](#).

When querying the OMX core, V550 searches for the following roles in the list of OpenMAX components:

H.264 decode

"video_decoder.avc"

JPEG decode

"video_decoder.mjpeg"

MPEG2 decode

"video_decoder.mpeg2"

MPEG4 decode

"video_decoder.mpeg4"

VC1 decode

"video_decoder.vc1"

VP8 decode

"video_decoder.vp8"

VP8 encode

"video_encoder.vp8"



To build example platforms containing V550, you must either install the TPIP package, or remove the dependency on FFmpeg and libvpx from the platform's `sgproj` file, by removing the line containing `v5xx.sgrepo`.

The model has the following limitations:

- No support for HEVC and RealVideo decoders.
- No support for 10-bit video output.
- No profiling support.
- No QoS support.
- Power and Test modes are modeled only as register state changes.

Ports for V550

Table 3-634: Ports

Name	Protocol	Type	Description
apb_s	PVBus	Slave	APB Slave port for register access.
axi_m	PVBus	Master	AXI master bus for memory accesses.
clk	ClockSignal	Slave	Master clock, typically 300MHz.
irq	Signal	Master	IRQ signal to host CPU.
reset	Signal	Slave	Reset signal.

Parameters for V550

AXI-data-width
AXI data width, logarithmic byte notation (3->64bit, 4->128 bit)
Type
int
Default value
0x4

enable-frame-rate-limiting
enable output rate control via the CLK port
Type
bool
Default value
0x0

fuse-disable-AFBC
disable AFBC support by fuse
Type
bool
Default value
0x0

fuse-disable-Real
disable RealVideo support by fuse

Type

bool

Default value

0x0

fuse-disable-VP8

disable VP8 support by fuse

Type

bool

Default value

0x0

ncores

Number of cores in the component

Type

int

Default value

0x1

omx-library-path

path to a user-provided OMX library; leave blank to use FFomaxIL

Type

string

Default value**supports-10bit**

component supports 10-bit content decoding

Type

bool

Default value

0x1

supports-VP8

component supports VP8

Type

bool

Default value

0x1

supports-encoding

component supports encoding

Type

bool

Default value

0x1

3.7 Peripheral components

This section describes the Peripheral components.

3.7.1 AudioOut_File

File based Audio Output for PL041_AACI. This model is written in LISA+.

Iris and MTI instances for AudioOut_File

This model has the following Iris instances:

Table 3-635: AudioOut_File Iris instances

InstanceName	ComponentName
AudioOut_File	AudioOut_File

AudioOut_File contains the following CADI targets:

- AudioOut_File

About AudioOut_File

This component implements an audio output that is suitable for use with the PL041_AACI component. It writes raw 16-bit 48KHz stereo audio data to a user-specified file.

We expect this component to have little effect on the performance of PV systems. AudioOut_File drains audio data at the rate that would be expected by software running in the simulation.

Ports for AudioOut_File

Table 3-636: Ports

Name	Protocol	Type	Description
audio	AudioControl	Slave	Audio input for a connection to a component such as the PL041_AACI.

Parameters for AudioOut_File

fname

Filename

Type

string

Default value

3.7.2 AudioOut_SDL

SDL based Audio Output for PL041_AACI. This model is written in LISA+.

Iris and MTI instances for AudioOut_SDL

This model has the following Iris instances:

Table 3-637: AudioOut_SDL Iris instances

InstanceName	ComponentName
AudioOut_SDL	AudioOut_SDL

AudioOut_SDL contains the following CADI targets:

- AudioOut_SDL

About AudioOut_SDL

AudioOut_SDL outputs audio using the host features of the Simple DirectMedia Layer (SDL) library.

This component results in SDL audio callbacks and might have a small impact on PV systems containing the component. It attempts to drain audio data at whatever rate is required to maintain smooth sound playback on the host PC. This might not match the data rate expected by applications running on the simulation.

Ports for AudioOut_SDL

Table 3-638: Ports

Name	Protocol	Type	Description
audio	AudioControl	Slave	Audio input for a connection to a component such as the PL041_AACI.

3.7.3 Base_PowerController

Base Platforms Power Controller. This model is written in LISA+.

Changes in 11.26.8

Ports added:

- delayed_ppu_transition_handler

Iris and MTI instances for Base_PowerController

This model has the following Iris instances:

Table 3-639: Base_PowerController Iris instances

InstanceName	ComponentName
Base_PowerController	Base_PowerController
Base_PowerController.busslave	PVBusSlave

InstanceName	ComponentName
Base_PowerController.timer_ppu_transition	ClockTimerThread
Base_PowerController.timer_ppu_transition.timer	ClockTimerThread64
Base_PowerController.timer_ppu_transition.timer.thread	SchedulerThread
Base_PowerController.timer_ppu_transition.timer.thread_event	SchedulerThreadEvent
Base_PowerController.timer_reset	ClockTimerThread
Base_PowerController.timer_reset.timer	ClockTimerThread64
Base_PowerController.timer_reset.timer.thread	SchedulerThread
Base_PowerController.timer_reset.timer.thread_event	SchedulerThreadEvent
Base_PowerController.utility_bus0	PVBusMaster
Base_PowerController.utility_bus1	PVBusMaster
Base_PowerController.utility_bus2	PVBusMaster
Base_PowerController.utility_bus3	PVBusMaster

This model has the following MTI trace components:

Table 3-640: Base_PowerController MTI instances

InstanceName	ComponentName
Base_PowerController	Base_PowerController
Base_PowerController.busslave	PVBusSlave
Base_PowerController.utility_bus0	PVBusMaster
Base_PowerController.utility_bus1	PVBusMaster
Base_PowerController.utility_bus2	PVBusMaster
Base_PowerController.utility_bus3	PVBusMaster

Base_PowerController contains the following CADI targets:

- Base_PowerController
- ClockTimerThread
- ClockTimerThread64
- SchedulerThread
- SchedulerThreadEvent

Ports for Base_PowerController

Table 3-641: Ports

Name	Protocol	Type	Description
cpuporeset[36]	Signal	Master	-
dbgnopwrdown[36]	Signal	Slave	-
delayed_ppu_transition_handler	TimerCallback	Slave	-
l2reset[4]	Signal	Master	-
pchannel_m[36]	PChannel	Master	-

Name	Protocol	Type	Description
pvbus_s	PVBus	Slave	-
standbywfi[36]	Signal	Slave	-
standbywfi12[4]	Signal	Slave	-
system_reset	Signal	Master	-
system_reset_req	Signal	Slave	-
timer_callback_handler	TimerCallback	Slave	Handle event notifications from the timer.
utility_bus_m[4]	PVBus	Master	-
wakerequest[36]	Signal	Slave	-

Parameters for Base_PowerController

Affinity-shifted

Whether core number is reflected in Affinity1 instead of Affinity0

Type

bool

Default value

0x0

CPU-affinities

Definition of which cores are attached to the control pins, as a comma separated list of affinity dotted quads

Type

string

Default value

0.0.0.0

CPU-available-mask

One bit per entry in CPU-affinities list, set zero if a CPU is wired up but actually not available

Type

int

Default value

0xffffffffffffffffffff

enable_lock_step

If lock step is enabled, the number of available cores get reduced to half

Type

bool

Default value

0x0

startup

Comma-separated list of cores (wildcards allowed) to be powered up at startup or system reset

Type

string

Default value

0.0.0.*

use_in_cluster_ppu

Set this to true if base power controller is connected to V9 core where in-cluster PPU is used, false, otherwise.

Type

bool

Default value

0x0

use_pchannel_for_threads

Set this to true if the pchannel is connected to cpus with thread support.

Type

bool

Default value

0x0

3.7.4 Cluster_Temperature_Sensor

Component to calculate the temperature value of all cores in a cluster. This model is written in C++.

Ports for Cluster_Temperature_Sensor

Table 3-642: Ports

Name	Protocol	Type	Description
cluster_powerdown_in	Signal	Slave	-
core_powerdown_in[16]	Signal	Slave	-
core_state_in[16]	ValueState	Slave	-
core_ticks_in[16]	InstructionCount	Slave	-
freq_in	ValueState	Slave	-
temperature_out	ValueState	Master	-

Parameters for Cluster_Temperature_Sensor

- A
- Temperature Coefficient

Type

string

Default value

0.5

B

Temperature Coefficient

Type

string

Default value

0.5

K

Temperature Coefficient

Type

int

Default value

0x32

MAX_FREQ

Maximum frequency (in Hz) at which each core can run

Type

int

Default value

0x77359400

NUM_CORES

Number of cores per cluster

Type

int

Default value

0x4

TAMB

Temperature Coefficient

Type

int

Default value

0x14

diagnostics

Diagnostics

Type

int

Default value

0x0

3.7.5 CombinedMessagingUnit

CMU - Combined MHU monolithic block. This model is written in LISA+.

Iris and MTI instances for CombinedMessagingUnit

This model has the following Iris instances:

Table 3-643: CombinedMessagingUnit Iris instances

InstanceName	ComponentName
CombinedMessagingUnit	CombinedMessagingUnit
CombinedMessagingUnit.host_to_local	MessageHandlingUnit
CombinedMessagingUnit.host_to_local.a_to_b_v2	MessageHandlingUnitV2
CombinedMessagingUnit.host_to_local.a_to_b_v3	MessageHandlingUnitV3
CombinedMessagingUnit.host_to_local.version_mapper_a_to_b_rec	PVBusMapper
CombinedMessagingUnit.host_to_local.version_mapper_a_to_b_snd	PVBusMapper
CombinedMessagingUnit.host_to_local_rcv_log	PVBusLogger
CombinedMessagingUnit.host_to_local_rcv_log.mapper	PVBusMapper
CombinedMessagingUnit.host_to_local_snd_log	PVBusLogger
CombinedMessagingUnit.host_to_local_snd_log.mapper	PVBusMapper
CombinedMessagingUnit.irq_rcv_combined_host_log	SignalLogger
CombinedMessagingUnit.irq_rcv_combined_local_log	SignalLogger
CombinedMessagingUnit.irq_snd_combined_host_log	SignalLogger
CombinedMessagingUnit.irq_snd_combined_local_log	SignalLogger
CombinedMessagingUnit.local_to_host	MessageHandlingUnit
CombinedMessagingUnit.local_to_host.a_to_b_v2	MessageHandlingUnitV2
CombinedMessagingUnit.local_to_host.a_to_b_v3	MessageHandlingUnitV3
CombinedMessagingUnit.local_to_host.version_mapper_a_to_b_rec	PVBusMapper
CombinedMessagingUnit.local_to_host.version_mapper_a_to_b_snd	PVBusMapper
CombinedMessagingUnit.local_to_host_rcv_log	PVBusLogger
CombinedMessagingUnit.local_to_host_rcv_log.mapper	PVBusMapper
CombinedMessagingUnit.local_to_host_snd_log	PVBusLogger
CombinedMessagingUnit.local_to_host_snd_log.mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-644: CombinedMessagingUnit MTI instances

InstanceName	ComponentName
CombinedMessagingUnit.host_to_local.a_to_b_v2	MessageHandlingUnitV2
CombinedMessagingUnit.host_to_local.a_to_b_v3	MessageHandlingUnitV3
CombinedMessagingUnit.host_to_local.version_mapper_a_to_b_rec	PVBusMapper
CombinedMessagingUnit.host_to_local.version_mapper_a_to_b_snd	PVBusMapper
CombinedMessagingUnit.host_to_local_rcv_log	PVBusLogger
CombinedMessagingUnit.host_to_local_rcv_log.mapper	PVBusMapper
CombinedMessagingUnit.host_to_local_snd_log	PVBusLogger
CombinedMessagingUnit.host_to_local_snd_log.mapper	PVBusMapper
CombinedMessagingUnit.irq_rcv_combined_host_log	SignalLogger
CombinedMessagingUnit.irq_rcv_combined_local_log	SignalLogger
CombinedMessagingUnit.irq_snd_combined_host_log	SignalLogger
CombinedMessagingUnit.irq_snd_combined_local_log	SignalLogger
CombinedMessagingUnit.local_to_host.a_to_b_v2	MessageHandlingUnitV2
CombinedMessagingUnit.local_to_host.a_to_b_v3	MessageHandlingUnitV3
CombinedMessagingUnit.local_to_host.version_mapper_a_to_b_rec	PVBusMapper
CombinedMessagingUnit.local_to_host.version_mapper_a_to_b_snd	PVBusMapper
CombinedMessagingUnit.local_to_host_rcv_log	PVBusLogger
CombinedMessagingUnit.local_to_host_rcv_log.mapper	PVBusMapper
CombinedMessagingUnit.local_to_host_snd_log	PVBusLogger
CombinedMessagingUnit.local_to_host_snd_log.mapper	PVBusMapper

CombinedMessagingUnit contains the following CADI targets:

- CombinedMessagingUnit
- MessageHandlingUnit
- MessageHandlingUnitV2
- MessageHandlingUnitV3
- PVBusLogger
- SignalLogger

Ports for CombinedMessagingUnit

Table 3-645: Ports

Name	Protocol	Type	Description
irq_rcv_combined_host	Signal	Master	-
irq_rcv_combined_local	Signal	Master	-
irq_snd_combined_host	Signal	Master	-
irq_snd_combined_local	Signal	Master	-
pvbus_s_rcv_host	PVBus	Slave	-

Name	Protocol	Type	Description
pvbus_s_rcv_local	PVBus	Slave	-
pvbus_s_snd_host	PVBus	Slave	-
pvbus_s_snd_local	PVBus	Slave	-
reset_in	Signal	Slave	-

Parameters for CombinedMessagingUnit

NUM_DB_CH

Number of doorbell channels

Type

int

Default value

0x1

NUM_FAST_CH

Number of fast channels

Type

int

Default value

0x1

diagnostics

Diagnostics 0==FATAL_ERROR -> 4==DEBUG

Type

int

Default value

0x2

fast_ch_group_int_enable

Fast Channel group interrupts enable, default=false

Type

bool

Default value

0x0

fast_ch_n_per_group

Fast Channel num channels per group, default=1

Type

int

Default value

0x1

fast_ch_num_groups

Fast Channel num of groups, default=1

Type

int

Default value

0x1

fast_ch_word_size

Fast Channel word size 32bit or 64bit, default=32

Type

int

Default value

0x20

host_to_local.a_to_b_v3.NUM_FIFO_CH

Number of FIFO Channels, default=1

Type

int

Default value

0x1

host_to_local.a_to_b_v3.auto_op_full

AutoOp mode - AutoOp(min) == false, AutoOp(full) == true - default: AutoOp(min)

Type

bool

Default value

0x0

host_to_local.a_to_b_v3.fast_ch_group_int_enable

Fast Channel group interrupts enable, default=false

Type

bool

Default value

0x0

host_to_local.a_to_b_v3.fifo_depth

Depth of the FIFO = fifo_depth + 1

Type

int

Default value

0x4

host_to_local.a_to_b_v3.m16ba_spt

Mailbox 16 bit access support to FIFO registers

Type

bool

Default value

0x0

host_to_local.a_to_b_v3.m32ba_spt

Mailbox 32 bit access support to FIFO registers

Type

bool

Default value

0x1

host_to_local.a_to_b_v3.m64ba_spt

Mailbox 64 bit access support to FIFO registers

Type

bool

Default value

0x0

host_to_local.a_to_b_v3.m8ba_spt

Mailbox 8 bit access support to FIFO registers

Type

bool

Default value

0x0

host_to_local.a_to_b_v3.monolithic

Monolithic or Distributed MHU - default: monolithic(true)

Type

bool

Default value

0x1

host_to_local.a_to_b_v3.p16ba_spt

Postbox 16 bit access support to FIFO registers

Type

bool

Default value

0x0

host_to_local.a_to_b_v3.p32ba_spt

Postbox 32 bit access support to FIFO registers

Type

bool

Default value

0x1

host_to_local.a_to_b_v3.p64ba_spt

Postbox 64 bit access support to FIFO registers

Type

bool

Default value

0x0

host_to_local.a_to_b_v3.p8ba_spt

Postbox 8 bit access support to FIFO registers

Type

bool

Default value

0x0

host_to_local.fast_ch_group_int_enable

Fast Channel group interrupts enable, default=false

Type

bool

Default value

0x0

host_to_local_rcv_log.trace_snoops

Enable tracing of ACE snoop requests

Type

bool

Default value

0x0

host_to_local_snd_log.trace_snoops

Enable tracing of ACE snoop requests

Type

bool

Default value

0x0

irq_rcv_combined_host_log.forward_signal

If true, trace signal and forward signal from 'in' to 'out'. If false, trace signal only without driving 'out' port

Type

bool

Default value

0x1

irq_rcv_combined_local_log.forward_signal

If true, trace signal and forward signal from 'in' to 'out'. If false, trace signal only without driving 'out' port

Type

bool

Default value

0x1

irq_snd_combined_host_log.forward_signal

If true, trace signal and forward signal from 'in' to 'out'. If false, trace signal only without driving 'out' port

Type

bool

Default value

0x1

irq_snd_combined_local_log.forward_signal

If true, trace signal and forward signal from 'in' to 'out'. If false, trace signal only without driving 'out' port

Type

bool

Default value

0x1

local_to_host.a_to_b_v3.NUM_FIFO_CH

Number of FIFO Channels, default=1

Type

int

Default value

0x1

local_to_host.a_to_b_v3.auto_op_full

AutoOp mode - AutoOp(min) == false, AutoOp(full) == true - default: AutoOp(min)

Type

bool

Default value

0x0

local_to_host.a_to_b_v3.fast_ch_group_int_enable

Fast Channel group interrupts enable, default=false

Type

bool

Default value

0x0

local_to_host.a_to_b_v3.fifo_depth

Depth of the FIFO = fifo_depth + 1

Type

int

Default value

0x4

local_to_host.a_to_b_v3.m16ba_spt

Mailbox 16 bit access support to FIFO registers

Type

bool

Default value

0x0

local_to_host.a_to_b_v3.m32ba_spt

Mailbox 32 bit access support to FIFO registers

Type

bool

Default value

0x1

local_to_host.a_to_b_v3.m64ba_spt

Mailbox 64 bit access support to FIFO registers

Type

bool

Default value

0x0

local_to_host.a_to_b_v3.m8ba_spt

Mailbox 8 bit access support to FIFO registers

Type

bool

Default value

0x0

local_to_host.a_to_b_v3.monolithic

Monolithic or Distributed MHU - default: monolithic(true)

Type

bool

Default value

0x1

local_to_host.a_to_b_v3.p16ba_spt

Postbox 16 bit access support to FIFO registers

Type

bool

Default value

0x0

local_to_host.a_to_b_v3.p32ba_spt

Postbox 32 bit access support to FIFO registers

Type

bool

Default value

0x1

local_to_host.a_to_b_v3.p64ba_spt

Postbox 64 bit access support to FIFO registers

Type

bool

Default value

0x0

local_to_host.a_to_b_v3.p8ba_spt

Postbox 8 bit access support to FIFO registers

Type

bool

Default value

0x0

local_to_host.fast_ch_group_int_enable

Fast Channel group interrupts enable, default=false

Type

bool

Default value

0x0

local_to_host_rcv_log.trace_snoops

Enable tracing of ACE snoop requests

Type

bool

Default value

0x0

local_to_host_snd_log.trace_snoops

Enable tracing of ACE snoop requests

Type

bool

Default value

0x0

major_version

MHU major version (default=2)

Type

int

Default value

0x2

mhu_arch_beta01

true = Aligns to MHUv3.beta01; false = Aligns to MHUv3.2

Type

bool

Default value

0x0

minor_version
MHU minor version (default=1)

Type
int

Default value
0x1

product_id
MHU part number

Type
int

Default value
0x0

3.7.6 DebugAccessPort

This model is written in C++.

Ports for DebugAccessPort

Table 3-646: Ports

Name	Protocol	Type	Description
ap_pvbus_m[2]	PVBus	Master	-
clock	ClockSignal	Slave	-
paddrdbg31	Signal	Master	-

Parameters for DebugAccessPort

ap0_has_debug_rom
Whether AP0 has a Debug ROM

Type
bool

Default value
0x0

ap0_rom_base_address
ROM base address for AP 0

Type
int

Default value
0x0

ap0_set_paddrdbg31

Set paddrdbg31 signal during accesses on AP0

Type

bool

Default value

0x0

ap1_has_debug_rom

Whether AP1 has a Debug ROM

Type

bool

Default value

0x0

ap1_rom_base_address

ROM base address for AP 1

Type

int

Default value

0x0

ap1_set_paddrdbg31

Set paddrdbg31 signal during accesses on AP1

Type

bool

Default value

0x0

3.7.7 DebugROM

This model is written in C++.

Ports for DebugROM

Table 3-647: Ports

Name	Protocol	Type	Description
paddrdbg31	Signal	Master	-
pvbus_s	PVBus	Slave	-

Parameters for DebugROM

customer_modified

Type

int

Default value

0x0

entry_0

Offset of component 0

Type

int

Default value

0x0

entry_1

Offset of component 1

Type

int

Default value

0x0

entry_10

Offset of component 10

Type

int

Default value

0x0

entry_11

Offset of component 11

Type

int

Default value

0x0

entry_12

Offset of component 12

Type

int

Default value

0x0

entry_13

Offset of component 13

Type

int

Default value

0x0

entry_14

Offset of component 14

Type

int

Default value

0x0

entry_15

Offset of component 15

Type

int

Default value

0x0

entry_16

Offset of component 16

Type

int

Default value

0x0

entry_17

Offset of component 17

Type

int

Default value

0x0

entry_18

Offset of component 18

Type

int

Default value

0x0

entry_19

Offset of component 19

Type

int

Default value

0x0

entry_2

Offset of component 2

Type

int

Default value

0x0

entry_20

Offset of component 20

Type

int

Default value

0x0

entry_21

Offset of component 21

Type

int

Default value

0x0

entry_22

Offset of component 22

Type

int

Default value

0x0

entry_23

Offset of component 23

Type

int

Default value

0x0

entry_24

Offset of component 24

Type

int

Default value

0x0

entry_25

Offset of component 25

Type

int

Default value

0x0

entry_26

Offset of component 26

Type

int

Default value

0x0

entry_27

Offset of component 27

Type

int

Default value

0x0

entry_28

Offset of component 28

Type

int

Default value

0x0

entry_29

Offset of component 29

Type

int

Default value

0x0

entry_3

Offset of component 3

Type

int

Default value

0x0

entry_30

Offset of component 30

Type

int

Default value

0x0

entry_31

Offset of component 31

Type

int

Default value

0x0

entry_32

Offset of component 32

Type

int

Default value

0x0

entry_33

Offset of component 33

Type

int

Default value

0x0

entry_34

Offset of component 34

Type

int

Default value

0x0

entry_35

Offset of component 35

Type

int

Default value

0x0

entry_36

Offset of component 36

Type

int

Default value

0x0

entry_37

Offset of component 37

Type

int

Default value

0x0

entry_38

Offset of component 38

Type

int

Default value

0x0

entry_39

Offset of component 39

Type

int

Default value

0x0

entry_4

Offset of component 4

Type

int

Default value

0x0

entry_40

Offset of component 40

Type

int

Default value

0x0

entry_41

Offset of component 41

Type

int

Default value

0x0

entry_42

Offset of component 42

Type

int

Default value

0x0

entry_43

Offset of component 43

Type

int

Default value

0x0

entry_44

Offset of component 44

Type

int

Default value

0x0

entry_45

Offset of component 45

Type

int

Default value

0x0

entry_46

Offset of component 46

Type

int

Default value

0x0

entry_47

Offset of component 47

Type

int

Default value

0x0

entry_48

Offset of component 48

Type

int

Default value

0x0

entry_49

Offset of component 49

Type

int

Default value

0x0

entry_5

Offset of component 5

Type

int

Default value

0x0

entry_50

Offset of component 50

Type

int

Default value

0x0

entry_51

Offset of component 51

Type

int

Default value

0x0

entry_52

Offset of component 52

Type

int

Default value

0x0

entry_53

Offset of component 53

Type

int

Default value

0x0

entry_54

Offset of component 54

Type

int

Default value

0x0

entry_55

Offset of component 55

Type

int

Default value

0x0

entry_56

Offset of component 56

Type

int

Default value

0x0

entry_57

Offset of component 57

Type

int

Default value

0x0

entry_58

Offset of component 58

Type

int

Default value

0x0

entry_59

Offset of component 59

Type

int

Default value

0x0

entry_6

Offset of component 6

Type

int

Default value

0x0

entry_60

Offset of component 60

Type

int

Default value

0x0

entry_61

Offset of component 61

Type

int

Default value

0x0

entry_62

Offset of component 62

Type

int

Default value

0x0

entry_63

Offset of component 63

Type

int

Default value

0x0

entry_7

Offset of component 7

Type

int

Default value

0x0

entry_8

Offset of component 8

Type

int

Default value

0x0

entry_9

Offset of component 9

Type

int

Default value

0x0

manufacturer_revision_number

Type
int

Default value
0x0

part_number

Type
int

Default value
0x0

revision

Type
int

Default value
0x0

3.7.8 DualClusterSystemConfigurationBlock

Dual Cluster System Configuration Block. This model is written in LISA+.

Iris and MTI instances for DualClusterSystemConfigurationBlock

This model has the following Iris instances:

Table 3-648: DualClusterSystemConfigurationBlock Iris instances

InstanceName	ComponentName
DualClusterSystemConfigurationBlock	DualClusterSystemConfigurationBlock
DualClusterSystemConfigurationBlock.pvbusslave	PVBusSlave

This model has the following MTI trace components:

Table 3-649: DualClusterSystemConfigurationBlock MTI instances

InstanceName	ComponentName
DualClusterSystemConfigurationBlock	DualClusterSystemConfigurationBlock
DualClusterSystemConfigurationBlock.pvbusslave	PVBusSlave

DualClusterSystemConfigurationBlock contains the following CADI targets:

- DualClusterSystemConfigurationBlock

Ports for DualClusterSystemConfigurationBlock

Table 3-650: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	-
cluster0_cfgend[4]	Signal	Master	-
cluster0_cfggte[4]	Signal	Master	-
cluster0_clusterid	Value	Master	-
cluster0_corereset[4]	Signal	Master	-
cluster0_cpuporeset[4]	Signal	Master	-
cluster0_cxreset[4]	Signal	Master	-
cluster0_eventi	Signal	Peer	-
cluster0_evento	Signal	Peer	-
cluster0_iminlen	Signal	Master	-
cluster0_l2reset	Signal	Master	-
cluster0_standbywfi[4]	Signal	Slave	-
cluster0_vinithi[4]	Signal	Master	-
cluster1_cfgend[4]	Signal	Master	-
cluster1_clusterid	Value	Master	-
cluster1_corereset[4]	Signal	Master	-
cluster1_cpuporeset[4]	Signal	Master	-
cluster1_eventi	Signal	Peer	-
cluster1_evento	Signal	Peer	-
cluster1_scureset	Signal	Master	-
cluster1_standbywfi[4]	Signal	Slave	-
cluster1_teinit[4]	Signal	Master	-
cluster1_vinithi[4]	Signal	Master	-
daughter_leds_state	ValueState	Master	-
daughter_user_switches	ValueState	Master	-
intgen[128]	Signal	Master	-
periphbase	Value_64	Master	-
periphbase_32	Value	Master	-
pvbus	PVBus	Slave	-
system_reset	Signal	Master	-
vgic_configuration_port	v7_VGIC_Configuration_Protocol	Master	-

Parameters for DualClusterSystemConfigurationBlock

CFG_ACTIVECLUSTER

Select which cluster will come out of reset coming out of power-on: bit[0] for primary cluster (Cortex-A15), bit[1] for secondary cluster (Cortex-A7). Value 0 is not allowed as it will hold both clusters in reset indefinitely!

Type

int

Default value

0x1

Cluster0IdOnPOReset

ClusterId for primary cluster (Cortex-A15) on power-on reset

Type

int

Default value

0x0

Cluster1IdOnPOReset

ClusterId for secondary cluster (Cortex-A7) on power-on reset

Type

int

Default value

0x1

DCSCB_PERIPHBASE

PERIPHBASE

Type

int

Default value

0x1e000000

DCS_AID

DCS_AID is the Auxiliary ID Register

Type

int

Default value

0x0

DCS_ID

The value returned by the DCS_ID register.

Type

int

Default value

0x41120000

DCS_ID_BUILD_NUMBER

DCS_ID build number

Type

int

Default value

0x1

DCS_LEDS

DCS_LEDS represents eight LEDs on the board that form an 8-bit value that can be r/w from the Dual Cluster System Configuration Block

Type

int

Default value

0x0

DCS_SW

DCS_SW represents eight switches on the board that form an 8-bit value that can be read from the Dual Cluster System Configuration Block

Type

int

Default value

0x0

FlipVGICWiringForCluster0AndCluster1

Flip the VGIC wiring round for cluster0 and cluster1. With this false, then cpu0 of cluster0 is cpu interface 0 on the VGIC. If this is true then cpu0 of cluster1 becomes cpu interface 0 on the VGIC.

Type

bool

Default value

0x0

INTGEN_INTS

Number of custom IRQs controlled by interrupt generator is $\text{INTGEN_INTS} * 32 + 32$.

Type

int

Default value

0x3

NumberOfCoresInCluster0

The number of cores in the primary cluster

Type

int

Default value

0x2

NumberOfCoresInCluster1

The number of cores in the secondary cluster

Type

int

Default value

0x2

ResetValueOfDaughterUserSwitches

Reset value of the user switches on the daughterboard

Type

int

Default value

0x0

stop_on_sequence_id

If non-zero the sequence_id of the SW trace mechanism on which to halt the simulator

Type

int

Default value

0x0

3.7.9 DummyAPB

DummyAPB. This model is written in LISA+.

Iris and MTI instances for DummyAPB

This model has the following Iris instances:

Table 3-651: DummyAPB Iris instances

InstanceName	ComponentName
DummyAPB	DummyAPB
DummyAPB.pvbuslave	PVBusSlave

This model has the following MTI trace components:

Table 3-652: DummyAPB MTI instances

InstanceName	ComponentName
DummyAPB.pvbuslave	PVBusSlave

DummyAPB contains the following CADI targets:

- DummyAPB

About DummyAPB

Use this dummy RAZ/WI APB device component to ensure that software does not receive aborts for accesses to devices that should be part of the system, but are not modeled.

For validation purposes it is useful to have dummy devices that are mostly RAZ/WI but return the correct value when you read ID registers. You can do that with this component in the following ways:

- Specify `periphid_24` for peripherals that follow the ARM pattern of having 12 ID registers at the top of an APB frame. For example:

```
periphid_24="04000000c2b00b000df005b1"
```

You need also to set `periph_framesize` to 4 or 64, depending on whether the peripheral has its registers in a 4KB or 64KB frame.

- Give a space-separated list of *offset:value* pairs in the `periphid_generic` parameter to define read-only values from particular offsets. For example:

```
periphid_generic="000:02468ace 1fc:13579bdf"
```

The number of hex digits used to specify the address is used to define the width of the address mask used. For example, `bc:02468ace` returns `02468ace` at reads from any address ending `bc`.

- Give a space-separated list of *offset:default-value* pairs in the `ram_generic` parameter to construct RAM. That is, the register at the relevant offset returns the default-value, but if changed, it returns the value that it is changed to.

Ports for DummyAPB

Table 3-653: Ports

Name	Protocol	Type	Description
pvbus_s	PVBus	Slave	Bus slave interface.

Parameters for DummyAPB

fail

Abort all accesses

Type

bool

Default value

0x0

failmsg

String to print when 'fail'=true and access occurred

Type

string

Default value**periph_framesize**

Size of frame (4/64, indicating if ID is at xFD0 or xFFD0)

Type

int

Default value

0xffffffffffffffff

periphid_24

24 hex digits for the 12 bytes of peripheral ID

Type

string

Default value**periphid_generic**

Set of space-separated offset:value pairs for dwords of ID

Type

string

Default value**ram_generic**

Set of space-separated offset:default pairs for writable dwords

Type

string

Default value**warn_once**

Warn once for the invalid read and write access.

Type

bool

Default value

0x1

3.7.10 ElfLoader

ELF loader component. This model is written in LISA+.

Iris and MTI instances for ElfLoader

This model has the following Iris instances:

Table 3-654: ElfLoader Iris instances

InstanceName	ComponentName
ElfLoader	ElfLoader
ElfLoader.pvbus_busmaster	PVBusMaster

This model has the following MTI trace components:

Table 3-655: ElfLoader MTI instances

InstanceName	ComponentName
ElfLoader.pvbus_busmaster	PVBusMaster

ElfLoader contains the following CADI targets:

- ElfLoader

About ElfLoader

ElfLoader provides an alternative method of loading ELF files into the system. It can load files in either of the following formats, or in gzip-compressed versions of them:

- ELF.
- Motorola S-Record.

Ports for ElfLoader

Table 3-656: Ports

Name	Protocol	Type	Description
pvbus_m	PVBus	Master	Master port for all memory accesses.
start_address	Value_64	Master	Provides a value reflecting the entry point of the last ELF image to be loaded.

Parameters for ElfLoader

elf

ELF file

Type

string

Default value

impdef_copy

DEPRECATED: Use realm_copy or root_copy parameters. load ELF file to implementation defined memory spaces, if load file is not specified.

Type

bool

Default value

0x0

lfile

load file for large address mapping

Type

string

Default value**ns_copy**

copy whole file to NS memory space

Type

bool

Default value

0x1

output_attributes_parameter_of_core

Encoding of various attributes on the bus.

Type

string

Default value

ExtendedID[54:39]=MPAM_PARTID, ExtendedID[38]=MPAM_SP[0],
ExtendedID[37]=MPAM_SP[1], UserFlags[31:16]=IMPDEF2

realm_copy

load ELF file to REALM memory spaces, if load file is not specified

Type

bool

Default value

0x0

root_copy

load ELF file to ROOT memory spaces, if load file is not specified

Type

bool

Default value
0x0

3.7.11 FlashLoader

A device that can preload a gzipped image into flash at startup. This model is written in C++.

Iris and MTI instances for FlashLoader
This model has the following Iris instances:

Table 3-657: FlashLoader Iris instances

InstanceName	ComponentName
FlashLoader	FlashLoader

FlashLoader contains the following CADI targets:

- FlashLoader

About FlashLoader

This component complements the IntelStrataFlashJ3 component by providing a means to initialize the contents of up to four Flash components in sequence from a single host flash image file.

Ports for FlashLoader

Table 3-658: Ports

Name	Protocol	Type	Description
flash_device0	FlashLoaderPort	Master	Used to program a flash device.
flash_device1	FlashLoaderPort	Master	Used to program a flash device.
flash_device2	FlashLoaderPort	Master	Used to program a flash device.
flash_device3	FlashLoaderPort	Master	Used to program a flash device.
warm_reset	Signal	Slave	Reset signal from external master.

Parameters for FlashLoader

Diagnostics

Diagnostics

Type

int

Default value

0x0

fname

Filename (Default '(none)' means: Do not load any file. An empty string will cause a warning.)

Type

string

Default value

(none)

fnameWrite

FilenameWrite (Default '(none)' means: Do not save any file. An empty string will cause a warning.)

Type

string

Default value

(none)

write_flash_after_reset

write_flash_after_reset

Type

bool

Default value

0x0

3.7.12 GICv3CommsLogger

Traces GICv3Comms activity. This model is written in LISA+.

Iris and MTI instances for GICv3CommsLogger

This model has the following Iris instances:

Table 3-659: GICv3CommsLogger Iris instances

InstanceName	ComponentName
GICv3CommsLogger	GICv3CommsLogger

This model has the following MTI trace components:

Table 3-660: GICv3CommsLogger MTI instances

InstanceName	ComponentName
GICv3CommsLogger	GICv3CommsLogger

GICv3CommsLogger contains the following CADI targets:

- GICv3CommsLogger

Ports for GICv3CommsLogger

Table 3-661: Ports

Name	Protocol	Type	Description
to_cpu	GICv3Comms	Master	To connect to CPU.
to_gic	GICv3Comms	Slave	To connect to GIC.

Parameters for GICv3CommsLogger

verbose

Print tracing information to attached debugger in addition to via MTI

Type

bool

Default value

0x0

3.7.13 GICv3CommsPVBUS

GICv3 Component for conversion between GICv3Comms protocol and PVBUS. This model is written in C++.

Iris and MTI instances for GICv3CommsPVBUS

This model has the following Iris instances:

Table 3-662: GICv3CommsPVBUS Iris instances

InstanceName	ComponentName
GICv3CommsPVBUS	GICv3CommsPVBUS
GICv3CommsPVBUS.bus_slave	PVBUSSlave

This model has the following MTI trace components:

Table 3-663: GICv3CommsPVBUS MTI instances

InstanceName	ComponentName
GICv3CommsPVBUS	GICv3CommsPVBUS
GICv3CommsPVBUS.bus_slave	PVBUSSlave

GICv3CommsPVBUS contains the following CADI targets:

- GICv3CommsPVBUS

Ports for GICv3CommsPVBUS

Table 3-664: Ports

Name	Protocol	Type	Description
axi_master_id_s[256]	Value	Slave	-

Name	Protocol	Type	Description
distributor_s[256]	GICv3Comms	Slave	-
pvbus_m	PVBus	Master	-
pvbus_s	PVBus	Slave	-

3.7.14 GICv3ProtocolChecker

GICv3 Component for command protocol checking. This model is written in C++.

Iris and MTI instances for GICv3ProtocolChecker

This model has the following Iris instances:

Table 3-665: GICv3ProtocolChecker Iris instances

InstanceName	ComponentName
GICv3ProtocolChecker	GICv3ProtocolChecker

This model has the following MTI trace components:

Table 3-666: GICv3ProtocolChecker MTI instances

InstanceName	ComponentName
GICv3ProtocolChecker	GICv3ProtocolChecker

GICv3ProtocolChecker contains the following CADI targets:

- GICv3ProtocolChecker

Ports for GICv3ProtocolChecker

Table 3-667: Ports

Name	Protocol	Type	Description
cpu_comms	GICv3Comms	Master	Master GICv3Comms port.
gicv3_comms	GICv3Comms	Slave	Slave GICv3Comms port.

Parameters for GICv3ProtocolChecker

cpu_interface_id

Cpu interface id to which this component is connected

Type

int

Default value

0x0

enable_protocol_checking

Enable/disable the protocol checking.

Type
bool

Default value
0x1

3.7.15 GUIPoll

Component providing a real-time periodic callback for GUI refresh. This model is written in C++.

About GUIPoll

Use a GUIPoll component as a subcomponent inside a LISA-based visualization component.

You can configure the period at which the GUIPoll component runs in milliseconds of real time, not simulation time. The component produces a `gui_callback()` signal at approximately this period, even when the simulation is paused.

You must implement the slave side of the `gui_callback()` signal in a LISA-based visualization component. Use this event to invoke the `visualisation::poll()` method to keep the GUI updated.

Ports for GUIPoll

Table 3-668: Ports

Name	Protocol	Type	Description
gui_callback	GUIPollCallback	Master	Sends callback requests to the visualization component.

Parameters for GUIPoll

delay_ms
GUI update period in ms

Type
int

Default value
0x32

has_gui
GUI is enabled

Type
bool

Default value
0x1

3.7.16 HostBridge

Host Socket Interface Component. This model is written in C++.

Iris and MTI instances for HostBridge

This model has the following Iris instances:

Table 3-669: HostBridge Iris instances

InstanceName	ComponentName
HostBridge	HostBridge

HostBridge contains the following CADI targets:

- HostBridge

About HostBridge

This component acts as a networking proxy for target NIC device models, to forward and receive ethernet packets to and from the host. Two kinds of proxy backend are integrated into this component:

- A host TAP/TUN-like network device, which is an ordinary TAP or MacVTap. This is the default.
- User mode networking, which emulates a built-in IP router and DHCP server to route traffic using the host user mode socket layer. To enable user mode networking, set the `userNetworking` parameter to true.

For more information see [1.10 User mode networking](#) on page 45.

HostBridge requires the following initialization sequence:

```
hostbridge.state.setValue(HostBridge::STATUS);  
hostbridge.state.setValue(HostBridge::S_UP);
```

To enable tracing of user-mode networking, which can help to debug networking issues, set the `FASTSIM_USERNET_DUMP` environment variable to any or all of the following values:

```
arpin,arpout,udpin,udpout,etherin,etherout,ipv4in,ipv4out,ipv4fragin,ipv4fragout,tcpin,tcpout
```

See also:

- [1.11.3 Configuring the networking environment for Microsoft Windows](#) on page 47
- [1.11.8 Configuring the networking environment for Linux](#) on page 53
- [Fast Models User Guide](#)

Ports for HostBridge

Table 3-670: Ports

Name	Protocol	Type	Description
eth	VirtualEthernet	Slave	-
state	ValueState_64	Slave	-

Parameters for HostBridge

interfaceName

Host Interface

Type

string

Default value

userNetOptions

Control options for UserNet TCP/IP (for internal use only, please do not use)

Type

string

Default value

userNetPorts

Listening ports to expose in user-mode networking

Type

string

Default value

userNetSubnet

Virtual subnet for user-mode networking

Type

string

Default value

172.20.51.0/24

userNetworking

Enable user-mode networking

Type

bool

Default value

0x0

3.7.17 HostSerialInterface

Component which provides access to the host serial interface. This model is written in LISA+.

Iris and MTI instances for HostSerialInterface

This model has the following Iris instances:

Table 3-671: HostSerialInterface Iris instances

InstanceName	ComponentName
HostSerialInterface	HostSerialInterface

HostSerialInterface contains the following CADI targets:

- HostSerialInterface

Ports for HostSerialInterface

Table 3-672: Ports

Name	Protocol	Type	Description
SerialData	SerialData	Slave	Serial data connection to export to host machine.

Parameters for HostSerialInterface

baud_rate
Baud rate override
Type
int
Default value
0x0

device
HW device to use
Type
string
Default value
/dev/ttyS0

3.7.18 Infra6_SRAM_ECC_RAS

ISD6.x SRAM ECC RAS. This model is written in C++.

Iris and MTI instances for Infra6_SRAM_ECC_RAS

This model has the following Iris instances:

Table 3-673: Infra6_SRAM_ECC_RAS Iris instances

InstanceName	ComponentName
Infra6_SRAM_ECC_RAS	Infra6_SRAM_ECC_RAS
Infra6_SRAM_ECC_RAS.PVBusModifier	PVBusMapper
Infra6_SRAM_ECC_RAS.pvbus_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-674: Infra6_SRAM_ECC_RAS MTI instances

InstanceName	ComponentName
Infra6_SRAM_ECC_RAS.PVBusModifier	PVBusMapper
Infra6_SRAM_ECC_RAS.pvbus_slave	PVBusSlave

Infra6_SRAM_ECC_RAS contains the following CADI targets:

- Infra6_SRAM_ECC_RAS

Ports for Infra6_SRAM_ECC_RAS

Table 3-675: Ports

Name	Protocol	Type	Description
fhi_irq_out	Signal	Master	-
pvbus_m	PVBus	Master	-
pvbus_reg_s	PVBus	Slave	-
pvbus_s	PVBus	Slave	-

Parameters for Infra6_SRAM_ECC_RAS

diagnostics

Diagnostics

Type

int

Default value

0x0

3.7.19 IntelStrataFlashJ3

Intel Strata Flash J3 LISA+ model. This model is written in LISA+.

Iris and MTI instances for IntelStrataFlashJ3

This model has the following Iris instances:

Table 3-676: IntelStrataFlashJ3 Iris instances

InstanceName	ComponentName
IntelStrataFlashJ3	IntelStrataFlashJ3
IntelStrataFlashJ3.map	PVBusMapper
IntelStrataFlashJ3.mbs	PVBusSlave
IntelStrataFlashJ3.rmbs	PVBusSlave

This model has the following MTI trace components:

Table 3-677: IntelStrataFlashJ3 MTI instances

InstanceName	ComponentName
IntelStrataFlashJ3.map	PVBusMapper
IntelStrataFlashJ3.mbs	PVBusSlave
IntelStrataFlashJ3.rmbs	PVBusSlave

IntelStrataFlashJ3 contains the following CADI targets:

- IntelStrataFlashJ3

About IntelStrataFlashJ3

This component is an efficient implementation of a NOR flash memory type device, an Intel StrataFlash Memory (J3). For information about Intel StrataFlash Memory (J3), see [Intel Download Center](#), [Intel StrataFlash Memory \(J3\) datasheet](#).

In normal usage, the device acts as Read Only Memory (ROM) whose contents can be determined either by programming using the flashloader port or by using standard flash programming software running on the model, such as the Arm® Firmware Suite.

The implementation of this component is approximately that of the Intel part in the VE development board. The component is effectively organized as a bank of two 16-bit Intel Flash components forming a 32-bit component that can be read or programmed in parallel. The component supports all hardware behavior except for:

- Protection register.
- Enhanced configuration register.
- Unique device identifier.
- One time programmable cells.
- Suspend/resume, which is silently ignored.
- Status interrupt line.

All block operations are atomic. This means that the status register state machine status bit always reads 1, ready.

In normal operation, this component has no user-visible registers, but you can read from it as if it is memory.

Programming it or changing the configuration requires a sequence of special write operations, see general flash programming documentation. The component supports Common Flash Interface query operations, which allow drivers to determine the properties of the flash memory.

**Note**

The model interprets all writes as requests to the programming state machine, and there are many state-machine states that do not support subsequent reads and return 0xdeaddead for them. Therefore, when simulating a ROM, use the `trapwrite=true` option.

Use the `diagnostics` parameter to select the level of diagnostic output:

Level 0

None.

Level 1

Report probable driver error operations:

- Unaligned operations that fault.
- Accesses that the state machine does not expect.
- Transitions of the state machine to unknown states.
- Writes to locked blocks and illegal lock commands.

Level 2

Report unimplemented and therefore ignored operations, and log lock commands.

Level 3

Warn if a flash write attempts to set bits (the write works if `unphysical_writes=true`).

Level 4

Log every read and write.

Ports for IntelStrataFlashJ3**Table 3-678: Ports**

Name	Protocol	Type	Description
flashloader	FlashLoaderPort	Slave	Can be used to load code/data into the flash.
mem_port	PVDevice	Slave	PVDevice port that is connect PVBUSlave port.
pdbus	PVBus	Slave	This is where the address and the data comes from the bus.

Parameters for IntelStrataFlashJ3**diagnostics**

Diagnostic level

Type

int

Default value

0x0

enable_read_status_logic

Enables logic to handle the status register reads as per the '3 Volt Intel StrataFlash Memory' specification

Type

bool

Default value

0x0

model_blocklock

Model per-block locking and set all the blocks to locked state on reset

Type

bool

Default value

0x0

size

Memory Size

Type

int

Default value

0x40000

trapwrite

Generate abort on write

Type

bool

Default value

0x0

unphysical_writes

Writes to flash are overwrite not AND

Type

bool

Default value

0x1

3.7.20 Interrupt_Router

Interrupt Router Registers. This model is written in C++.

Ports for Interrupt_Router

Table 3-679: Ports

Name	Protocol	Type	Description
lockdown	Signal	Slave	-
out_interrupts0[64]	Signal	Master	-
out_interrupts1[64]	Signal	Master	-
out_interrupts10[64]	Signal	Master	-
out_interrupts11[64]	Signal	Master	-
out_interrupts12[64]	Signal	Master	-
out_interrupts13[64]	Signal	Master	-
out_interrupts14[64]	Signal	Master	-
out_interrupts15[64]	Signal	Master	-
out_interrupts2[64]	Signal	Master	-
out_interrupts3[64]	Signal	Master	-
out_interrupts4[64]	Signal	Master	-
out_interrupts5[64]	Signal	Master	-
out_interrupts6[64]	Signal	Master	-
out_interrupts7[64]	Signal	Master	-
out_interrupts8[64]	Signal	Master	-
out_interrupts9[64]	Signal	Master	-
pvbus_s	PVBus	Slave	-
reset_signal	Signal	Slave	-
shared_interrupt[428]	Signal	Slave	-
tamper_interrupt	Signal	Master	-

Parameters for Interrupt_Router

diagnostics

Diagnostics to indicate debug level

Type

int

Default value

0x0

ici_dst

Interrupt Controller Destination The values range from 0 to $(2^{16} - 1)$. The bit representation of these values indicate the ICIs possible for each indexed shared interrupt to be routed to

rw_access
Stream ID of master with Read Write access

Type
int

Default value
0x0

3.7.21 LS64TestingFIFO

FIFO peripheral supporting LS64 accesses for testing purposes. This model is written in LISA+.

Iris and MTI instances for LS64TestingFIFO

This model has the following Iris instances:

Table 3-680: LS64TestingFIFO Iris instances

InstanceName	ComponentName
LS64TestingFIFO	LS64TestingFIFO
LS64TestingFIFO.pvbuslave	PVBusSlave

This model has the following MTI trace components:

Table 3-681: LS64TestingFIFO MTI instances

InstanceName	ComponentName
LS64TestingFIFO.pvbuslave	PVBusSlave

LS64TestingFIFO contains the following CADI targets:

- LS64TestingFIFO

About LS64TestingFIFO

LS64TestingFIFO is a LISA component for testing the FEAT_LS64 architectural feature. It accepts ST64B instructions and places the supplied data into a configurable buffer. LD64B instructions can then read this data out of the FIFO. The value returned can configurably be bitwise inverted.

It also supports the ST64BV variants where success and failure are reported by a return result rather than by transaction success and failure.

Ports for LS64TestingFIFO

Table 3-682: Ports

Name	Protocol	Type	Description
pvbus_s	PVBus	Slave	Bus subordinate interface.

Parameters for LS64TestingFIFO

buffer_size
The number of 64-byte slots in the FIFO
Type
int
Default value
0x2
op_type
The operation performed on the 64-byte transaction data 0 - None, 1 - Bitwise Negate
Type
int
Default value
0x1

3.7.22 MemoryMappedGenericTimer

ARM Generic Timer. This model is written in LISA+.

Iris and MTI instances for MemoryMappedGenericTimer

This model has the following Iris instances:

Table 3-683: MemoryMappedGenericTimer Iris instances

InstanceName	ComponentName
MemoryMappedGenericTimer	MemoryMappedGenericTimer
MemoryMappedGenericTimer.busbase0	PVBusSlave
MemoryMappedGenericTimer.busbase1	PVBusSlave
MemoryMappedGenericTimer.busbase10	PVBusSlave
MemoryMappedGenericTimer.busbase11	PVBusSlave
MemoryMappedGenericTimer.busbase12	PVBusSlave
MemoryMappedGenericTimer.busbase13	PVBusSlave
MemoryMappedGenericTimer.busbase14	PVBusSlave
MemoryMappedGenericTimer.busbase15	PVBusSlave
MemoryMappedGenericTimer.busbase2	PVBusSlave
MemoryMappedGenericTimer.busbase3	PVBusSlave
MemoryMappedGenericTimer.busbase4	PVBusSlave
MemoryMappedGenericTimer.busbase5	PVBusSlave
MemoryMappedGenericTimer.busbase6	PVBusSlave
MemoryMappedGenericTimer.busbase7	PVBusSlave
MemoryMappedGenericTimer.busbase8	PVBusSlave

InstanceName	ComponentName
MemoryMappedGenericTimer.busbase9	PVBusSlave
MemoryMappedGenericTimer.busctlbase	PVBusSlave

This model has the following MTI trace components:

Table 3-684: MemoryMappedGenericTimer MTI instances

InstanceName	ComponentName
MemoryMappedGenericTimer.busbase0	PVBusSlave
MemoryMappedGenericTimer.busbase1	PVBusSlave
MemoryMappedGenericTimer.busbase10	PVBusSlave
MemoryMappedGenericTimer.busbase11	PVBusSlave
MemoryMappedGenericTimer.busbase12	PVBusSlave
MemoryMappedGenericTimer.busbase13	PVBusSlave
MemoryMappedGenericTimer.busbase14	PVBusSlave
MemoryMappedGenericTimer.busbase15	PVBusSlave
MemoryMappedGenericTimer.busbase2	PVBusSlave
MemoryMappedGenericTimer.busbase3	PVBusSlave
MemoryMappedGenericTimer.busbase4	PVBusSlave
MemoryMappedGenericTimer.busbase5	PVBusSlave
MemoryMappedGenericTimer.busbase6	PVBusSlave
MemoryMappedGenericTimer.busbase7	PVBusSlave
MemoryMappedGenericTimer.busbase8	PVBusSlave
MemoryMappedGenericTimer.busbase9	PVBusSlave
MemoryMappedGenericTimer.busctlbase	PVBusSlave

MemoryMappedGenericTimer contains the following CADI targets:

- MemoryMappedGenericTimer

Ports for MemoryMappedGenericTimer

Table 3-685: Ports

Name	Protocol	Type	Description
cntpsirq[8]	Signal	Master	-
cntvalueb	CounterInterface	Slave	For connection to MemoryMappedCounterModule.
pvbus_base_s[8]	PVBus	Slave	-
pvbus_ctlbase_s	PVBus	Slave	-
pvbus_el0base_s[8]	PVBus	Slave	-
timer_reset	Signal	Slave	Reset.

Parameters for MemoryMappedGenericTimer

bypass_ctlbase

Bypass CNTBase Access Control. Enable if only timer frame feature is required without CNTBase access control

Type

bool

Default value

0x0

cntel0acr_implemented

A bit-field of 8 bits, where bit {n} enables CNTELOACR for timer frame {n}

Type

int

Default value

0x0

diagnostics

Diagnostics

Type

int

Default value

0x0

frame_security

Hard-wired/configurable security for frames (N/S/X, one character per timer frame)

Type

string

Default value

num_timers

Number of timer frames

Type

int

Default value

0x1

3.7.23 MemoryMappedGenericTimer_cpp

ARM Generic Timer. This model is written in C++.

Iris and MTI instances for MemoryMappedGenericTimer_cpp

This model has the following Iris instances:

Table 3-686: MemoryMappedGenericTimer_cpp Iris instances

InstanceName	ComponentName
MemoryMappedGenericTimer_cpp	MemoryMappedGenericTimer
MemoryMappedGenericTimer_cpp.PVBusSlave	PVBusSlave

This model has the following MTI trace components:

Table 3-687: MemoryMappedGenericTimer_cpp MTI instances

InstanceName	ComponentName
MemoryMappedGenericTimer_cpp.PVBusSlave	PVBusSlave

MemoryMappedGenericTimer_cpp contains the following CADI targets:

- MemoryMappedGenericTimer

Ports for MemoryMappedGenericTimer_cpp

Table 3-688: Ports

Name	Protocol	Type	Description
cntpsirq[8]	Signal	Master	-
cntvalueb	CounterInterface	Slave	-
pvbus_base_s[8]	PVBus	Slave	-
pvbus_ctlbase_s	PVBus	Slave	-
pvbus_el0base_s[8]	PVBus	Slave	-
timer_reset	Signal	Slave	-

Parameters for MemoryMappedGenericTimer_cpp

bypass_ctlbase

Bypass CNTBase Access Control. Enable if only timer frame feature is required without CNTBase access control

Type

bool

Default value

0x0

cntel0acr_implemented

A bit-field of 8 bits, where bit {n} enables CNTELOACR for timer frame {n}

Type

int

Default value

0x0

diagnostics

Diagnostics

Type

int

Default value

0x0

frame_security

Hard-wired/configurable security for frames (N/S/X, one character per timer frame)

Type

string

Default value

num_timers

Number of timer frames

Type

int

Default value

0x1

3.7.24 MemoryMappedGenericWatchdog

ARM Generic Watchdog. This model is written in LISA+.

Iris and MTI instances for MemoryMappedGenericWatchdog

This model has the following Iris instances:

Table 3-689: MemoryMappedGenericWatchdog Iris instances

InstanceName	ComponentName
MemoryMappedGenericWatchdog	MemoryMappedGenericWatchdog
MemoryMappedGenericWatchdog.busctlbase	PVBusSlave
MemoryMappedGenericWatchdog.busrefbase	PVBusSlave

This model has the following MTI trace components:

Table 3-690: MemoryMappedGenericWatchdog MTI instances

InstanceName	ComponentName
MemoryMappedGenericWatchdog.busctlbase	PVBusSlave
MemoryMappedGenericWatchdog.busrefbase	PVBusSlave

MemoryMappedGenericWatchdog contains the following CADI targets:

- MemoryMappedGenericWatchdog

About MemoryMappedGenericWatchdog

This is a high-level watchdog that generates two interrupts rather than an interrupt followed by a reset.

Ports for MemoryMappedGenericWatchdog

Table 3-691: Ports

Name	Protocol	Type	Description
cntvalueb	CounterInterface	Slave	For connection to MemoryMappedCounterModule.
ctl_pvbus_s	PVBus	Slave	Access to control frame.
ref_pvbus_s	PVBus	Slave	Access to refresh frame.
reset_in	Signal	Slave	Reset.
WS0	Signal	Master	Set when watchdog expired for the first time.
WS1	Signal	Master	Set when watchdog expired for the second time.

Parameters for MemoryMappedGenericWatchdog

NONSECURE

Non-Secure

Type

bool

Default value

0x0

arch_version

Architecture version. Available 0 and 1

Type

int

Default value

0x0

diagnostics

Diagnostics

Type

int

Default value

0x0

product_id

Product Identifier

Type

int

Default value

0x0

3.7.25 MemoryMappedGenericWatchdog_cpp

ARM Generic Watchdog. This model is written in C++.

Iris and MTI instances for MemoryMappedGenericWatchdog_cpp

This model has the following Iris instances:

Table 3-692: MemoryMappedGenericWatchdog_cpp Iris instances

InstanceName	ComponentName
MemoryMappedGenericWatchdog_cpp	MemoryMappedGenericWatchdog
MemoryMappedGenericWatchdog_cpp.PVBusSlave	PVBusSlave

This model has the following MTI trace components:

Table 3-693: MemoryMappedGenericWatchdog_cpp MTI instances

InstanceName	ComponentName
MemoryMappedGenericWatchdog_cpp.PVBusSlave	PVBusSlave

MemoryMappedGenericWatchdog_cpp contains the following CADI targets:

- MemoryMappedGenericWatchdog

Ports for MemoryMappedGenericWatchdog_cpp

Table 3-694: Ports

Name	Protocol	Type	Description
cntvalueb	CounterInterface	Slave	-
ctl_pvbus_s	PVBus	Slave	-
ref_pvbus_s	PVBus	Slave	-
reset_in	Signal	Slave	-
WS0	Signal	Master	-
WS1	Signal	Master	-

Parameters for MemoryMappedGenericWatchdog_cpp

NONSECURE
Non-Secure
Type
bool
Default value
0x0
arch_version
Architecture version. Available 0 and 1
Type
int
Default value
0x0
diagnostics
Diagnostics
Type
int
Default value
0x0
product_id
Product Identifier
Type
int
Default value
0x0

3.7.26 NonVolatileCounter

Trusted Non-Volatile Counter unit. This model is written in LISA+.

Iris and MTI instances for NonVolatileCounter

This model has the following Iris instances:

Table 3-695: NonVolatileCounter Iris instances

InstanceName	ComponentName
NonVolatileCounter	NonVolatileCounter
NonVolatileCounter.pvbuslave	PVBusSlave

This model has the following MTI trace components:

Table 3-696: NonVolatileCounter MTI instances

InstanceName	ComponentName
NonVolatileCounter	NonVolatileCounter
NonVolatileCounter.pvbuslave	PVBusSlave

NonVolatileCounter contains the following CADI targets:

- NonVolatileCounter

Ports for NonVolatileCounter

Table 3-697: Ports

Name	Protocol	Type	Description
pvbus_s	PVBus	Slave	Bus slave interface.

Parameters for NonVolatileCounter

diagnostics

Diagnostics

Type

int

Default value

0x0

rst_non_tz_fw_cnt

Value of NON_TZ_FW_CNT at reset

Type

int

Default value

0x0

rst_tz_fw_cnt

Value of TZ_FW_CNT at reset

Type

int

Default value

0x0

secure

Instantiate model as Secure (1) or NS (0)

Type

bool

Default value

0x1

version

Version of the model functionality. Valid values are r0 and r1.

Type

string

Default value

r0

3.7.27 PCIeATC

This model is written in C++.

Iris and MTI instances for PCIeATC

This model has the following Iris instances:

Table 3-698: PCIeATC Iris instances

InstanceName	ComponentName
PCIeATC	validation_atc
PCIeATC.ExportTest.PCIeATC.mapper	PVBusMapper
PCIeATC.pvbus_id_routed_s[0]	PVBusSlave

This model has the following MTI trace components:

Table 3-699: PCIeATC MTI instances

InstanceName	ComponentName
PCIeATC	atc
PCIeATC.ExportTest.PCIeATC.mapper	PVBusMapper
PCIeATC.pvbus_id_routed_s[0]	PVBusSlave

PCIeATC contains the following CADI targets:

- validation_atc

Ports for PCIeATC

Table 3-700: Ports

Name	Protocol	Type	Description
atc	PCIeATC_get_if	Slave	-
disable_PRI_and_set_RF	Signal	Master	This is pulsed (set, then clear) when a condition occurs that causes a Response Failure. The correct response of the PCIe device is to disable PRI and to set the RF bit in the PRI header.

Name	Protocol	Type	Description
identify	SMMUv3AEMIdentifyProtocol	Master	The user has a chance to determine how the substreamid is extracted from the transactions received on pvbus_s by using this port. If it is unimplemented then the ATC will use the default policy identified in SMMUv3_FOR_PCIE.lisa <201601041554/>
pvbus_id_routed_s	PVBus	Slave	-
pvbus_m	PVBus	Master	-
pvbus_s	PVBus	Slave	-
uprgi	Signal	Master	This is pulsed (set, then clear) when an Unrecognised PRG Index is received. In a real PCIe device this would set the UPRGI bit in the PRI header.

Parameters for PCIeATC

atc_size

The maximum number of ATC entries. 0 is effectively a large number.

Type

int

Default value

0x0

seed

Seed for a random number generator.

Type

int

Default value

0x12345678

3.7.28 PLLClockControl

Clock Rate Control. This model is written in C++.

Iris and MTI instances for PLLClockControl

This model has the following Iris instances:

Table 3-701: PLLClockControl Iris instances

InstanceName	ComponentName
PLLClockControl	PLLClockControl
PLLClockControl.ClockDivider	ClockDivider

This model has the following MTI trace components:

Table 3-702: PLLClockControl MTI instances

InstanceName	ComponentName
PLLClockControl.ClockDivider	ClockDivider

PLLClockControl contains the following CADI targets:

- PLLClockControl

Ports for PLLClockControl

Table 3-703: Ports

Name	Protocol	Type	Description
clk_en	Signal	Slave	-
clk_in	ClockSignal	Slave	-
clk_out	ClockSignal	Master	-
clk_rate	ClockRateControl	Slave	-
clk_sel	Value	Slave	-
lock	Signal	Master	-
refclk_in	ClockSignal	Slave	-
unlock	Signal	Master	-

Parameters for PLLClockControl

diagnostics

Diagnostics

Type

int

Default value

0x0

3.7.29 PPUMultiThreadModeSwitch

PPU mode switch between single-thread mode and multi-thread mode. Support up to 8 cores and thread number per core is no more than 2. This model is written in C++.

Ports for PPUMultiThreadModeSwitch

Table 3-704: Ports

Name	Protocol	Type	Description
pchannel_from_ppu_s[8]	PChannel	Slave	-
pchannel_to_cpu_m[8]	PChannel	Master	-
wakerequest_from_gic_s[16]	Signal	Slave	-
wakerequest_to_ppu_m[8]	Signal	Master	-

Parameters for PPUMultiThreadModeSwitch

mt_mode	Multi-threaded mode
Type	bool
Default value	0x0

3.7.30 PS2Keyboard

Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component. This model is written in LISA+.

Iris and MTI instances for PS2Keyboard

This model has the following Iris instances:

Table 3-705: PS2Keyboard Iris instances

InstanceName	ComponentName
PS2Keyboard	PS2Keyboard
PS2Keyboard.ps2_clocktimer	ClockTimerThread
PS2Keyboard.ps2_clocktimer.timer	ClockTimerThread64
PS2Keyboard.ps2_clocktimer.timer.thread	SchedulerThread
PS2Keyboard.ps2_clocktimer.timer.thread_event	SchedulerThreadEvent

PS2Keyboard contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- PS2Keyboard
- SchedulerThread
- SchedulerThreadEvent

About PS2Keyboard

This component translates a stream of key press information into appropriate PS/2 serial data. The key press data stream must be provided by another component such as a visualization component.

Ports for PS2Keyboard

Table 3-706: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Clock signal to rate at which PS2Data signals are generated.

Name	Protocol	Type	Description
keyboard	KeyboardStatus	Slave	Receives keyboard input from, for example, the Visualisation component.
ps2	PS2Data	Master	Connection to the PS/2 controller, for example, the PL050_KMI.

3.7.31 PS2Mouse

Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component. This model is written in LISA+.

Iris and MTI instances for PS2Mouse

This model has the following Iris instances:

Table 3-707: PS2Mouse Iris instances

InstanceName	ComponentName
PS2Mouse	PS2Mouse
PS2Mouse.ps2_clocktimer	ClockTimerThread
PS2Mouse.ps2_clocktimer.timer	ClockTimerThread64
PS2Mouse.ps2_clocktimer.timer.thread	SchedulerThread
PS2Mouse.ps2_clocktimer.timer.thread_event	SchedulerThreadEvent

PS2Mouse contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- PS2Mouse
- SchedulerThread
- SchedulerThreadEvent

About PS2Mouse

This component implements the PS/2 register interface of a PS/2 style mouse. The mouse movement and button press data must come from another component such as a visualization component.

See also [1.14 Visualisation library](#) on page 59.

Ports for PS2Mouse

Table 3-708: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Clock signal to rate at which PS2Data signals are generated.
mouse	MouseStatus	Slave	Receives keyboard input from, for example, the Visualisation component.
ps2	PS2Data	Master	Connection to the PS/2 controller, for example the PL050_KMI.

3.7.32 PVBusGICv3Comms

GICv3 Component for conversion between GICv3Comms protocol and PVBus. This model is written in C++.

Iris and MTI instances for PVBusGICv3Comms

This model has the following Iris instances:

Table 3-709: PVBusGICv3Comms Iris instances

InstanceName	ComponentName
PVBusGICv3Comms	PVBusGICv3Comms
PVBusGICv3Comms.bus_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-710: PVBusGICv3Comms MTI instances

InstanceName	ComponentName
PVBusGICv3Comms	PVBusGICv3Comms
PVBusGICv3Comms.bus_slave	PVBusSlave

PVBusGICv3Comms contains the following CADI targets:

- PVBusGICv3Comms

Ports for PVBusGICv3Comms

Table 3-711: Ports

Name	Protocol	Type	Description
axi_master_id_m[256]	Value	Master	-
distributor_m[256]	GICv3Comms	Master	-
pvbus_m	PVBus	Master	-
pvbus_s	PVBus	Slave	-

3.7.33 PVMetaDataController

A simulation-only (not in hardware) component that can service metadata requests for transactions on PVBus. This model is written in C++.

About PVMetaDataController

PVMetaDataController provides the tag storage for Memory Tagging Extension versions FEAT_MTE2, FEAT_MTE3, and any later versions. It must be enabled and placed upstream of a RAM device for the feature to work for transactions routed to that RAM device. It might also be used to enable other future architectural features.

To enable `PVMetaDataController`, set the `is_enabled` parameter to true.

Ports for PVMetaDataController

Table 3-712: Ports

Name	Protocol	Type	Description
<code>pvbus_m</code>	PVBus	Master	-
<code>pvbus_s</code>	PVBus	Slave	-

Parameters for PVMetaDataController

`init_value`

Initialize metadata memory with this value. If one of `init_values_json` or `init_values_json_file` is specified, this value applies only to any metadata not specified in the JSON.

Type

int

Default value

0xd

`init_values_json`

A JSON value describing initial metadata values. Mutually exclusive with `init_values_json_file`. The format is as follows: { "regions": [{ "begin": 0x0, "end": 0x10000, "mte_tag": 0xa }, { "begin": 0x20000, "end": 0x50000, "mte_tag": 0xc }] }

Type

string

Default value

`init_values_json_file`

Path to a JSON file with initial metadata values. Mutually exclusive with `init_values_json`. The format is as follows: { "regions": [{ "begin": 0x0, "end": 0x10000, "mte_tag": 0xa }, { "begin": 0x20000, "end": 0x50000, "mte_tag": 0xc }] }

Type

string

Default value

`is_enabled`

If false, disables the MetaData controller functionality, and makes the component invisible to passing transactions.

Type

bool

Default value

0x0

mte_tag_carveout_json

JSON string that specifies the PA range of the tag carveout regions and the beginning of the PA range for which they provide tag storage. If `pa_regions_with_metadata_storage` defines which regions can have metadata, the tag carveout regions cannot overlap them, and each tagged region must be entirely covered by one of them. The block size must be ≥ 64 bytes and a power of 2, defaulting to 4KiB. The maximum block size supported is 4KiB. The carveout region size must be ≥ 4 KiB and a power of 2, and determines the size of the corresponding tagged region. { "regions": [{ "begin": 0x0, "tag_carveout_region": [0xfffff00000, 0xfffff00fff] }, { "begin": 0x20000, "tag_carveout_region": [0xfffff01000, 0xfffff01fff], "block_size": 0x100 }, { "begin": 0x100000, "tag_carveout_region": [0xfffff08000, 0xfffff0Bfff], "block_size": 0x2000 },] }

Type

string

Default value**mte_tag_carveout_json_file**

Path to a JSON file that specifies the PA range of the tag carveout regions with the same format as `mte_tag_carveout_json`. Only one of `mte_tag_carveout_json` and `mte_tag_carveout_json_file` can be used.

Type

string

Default value**mte_tag_carveout_tag_order**

Order of the tags within the MTE tag carveout blocks. This can be little-endian (same order as the corresponding tagged data) or big-endian (reverse order). The parameter accepts both '-' and '_', so 'little-endian', 'big-endian', 'little_endian' and 'big_endian' are all valid. THIS PARAMETER HAS NO FUNCTIONALITY AT THE MOMENT.

Type

string

Default value

little-endian

pa_regions_with_metadata_storage

Specify the address region where the metadata storage is available for each PAS in a JSON format. If the PAS does not have a region specified, the PAS has metadata storage for all of the space. The regions are defined by begin and end_incl addresses. Example: { "ns": [0xa0000000, 0xa0000fff], "s": [0xb0000000, 0xb0000fff], "rl": [0xc0000000, 0xc0000fff], "rt": [0xd0000000, 0xd0000fff] } ns: non-secure, s: secure, rl: realm, rt: root

Type

string

Default value

3.7.34 PchannelListener

Provides a dummy PChannel device to accept all request. This model is written in C++.

Ports for PchannelListener

Table 3-713: Ports

Name	Protocol	Type	Description
dev_pchannel_s	PChannel	Slave	-

Parameters for PchannelListener

diagnostics

Diagnostics.

Type

int

Default value

0x0

3.7.35 RAMDevice

RAM device, can be dynamic or static ram. This model is written in LISA+.

Iris and MTI instances for RAMDevice

This model has the following Iris instances:

Table 3-714: RAMDevice Iris instances

InstanceName	ComponentName
RAMDevice	RAMDevice
RAMDevice.bus_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-715: RAMDevice MTI instances

InstanceName	ComponentName
RAMDevice	RAMDevice
RAMDevice.bus_slave	PVBusSlave

RAMDevice contains the following CADI targets:

- RAMDevice

About RAMDevice

As a generic device, this component does not have a hardware revision code.

Ports for RAMDevice

Table 3-716: Ports

Name	Protocol	Type	Description
pvbus	PVBus	Slave	Bus slave interface.

Parameters for RAMDevice

enable_atomic_ops

Supports Atomic Operations

Type

bool

Default value

0x0

fill1

Fill pattern 1, initialise memory at start of simulation with alternating fill1, fill2 pattern

Type

int

Default value

0x0

fill2

Fill pattern 2, initialise memory at start of simulation with alternating fill1, fill2 pattern

Type

int

Default value

0x0

read_latency

Memory read latency (ps/byte)

Type

int

Default value

0x0

size

Memory Size

Type

int

Default value

0x100000000

write_latency

Memory write latency (ps/byte)

Type

int

Default value

0x0

3.7.36 ROM

Simple ROM device. This model is written in LISA+.

Iris and MTI instances for ROM

This model has the following Iris instances:

Table 3-717: ROM Iris instances

InstanceName	ComponentName
ROM	ROM
ROM.bus_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-718: ROM MTI instances

InstanceName	ComponentName
ROM.bus_slave	PVBusSlave

ROM contains the following CADI targets:

- ROM

Ports for ROM

Table 3-719: Ports

Name	Protocol	Type	Description
pvbus	PVBus	Slave	-

Parameters for ROM

abort_writes

Abort writes instead of ignoring them

Type

bool

Default value

0x0

log2_size
Log2 size (bytes) e.g. 20 is 1 MiB

Type
int

Default value
0x14

raw_image
Raw image file to load at init time

Type
string

Default value

3.7.37 RSS_Integ_Regs

RSS Integration Layer Registers. This model is written in C++.

Iris and MTI instances for RSS_Integ_Regs

This model has the following Iris instances:

Table 3-720: RSS_Integ_Regs Iris instances

InstanceName	ComponentName
RSS_Integ_Regs	RSS_Integration_Registers
RSS_Integ_Regs.ClockDivider	ClockDivider
RSS_Integ_Regs.PVBusSlave	PVBusSlave

This model has the following MTI trace components:

Table 3-721: RSS_Integ_Regs MTI instances

InstanceName	ComponentName
RSS_Integ_Regs.ClockDivider	ClockDivider
RSS_Integ_Regs.PVBusSlave	PVBusSlave

RSS_Integ_Regs contains the following CADI targets:

- RSS_Integration_Registers

Ports for RSS_Integ_Regs

Table 3-722: Ports

Name	Protocol	Type	Description
EXTMCPRESETn	Signal	Master	-
EXTSCPRESETn	Signal	Master	-

Name	Protocol	Type	Description
MCP_ATU_AP	Signal	Master	-
MCP_RAS_ERR_CLEAR	Signal	Master	-
pvbus_s	PVBus	Slave	-
REFCLK	ClockSignal	Slave	-
reset_in	Signal	Slave	-
RSSCORECLK	ClockSignal	Master	-
SCP_ATU_AP	Signal	Master	-
SCP_RAS_ERR_CLEAR	Signal	Master	-
SYSPLLCLK	ClockSignal	Slave	-

Parameters for RSS_Integ_Regs

chip_id

RSS Integration Register Chip ID

Type

int

Default value

0x0

diagnostics

RSS Integration Registers Diagnostics

Type

int

Default value

0x0

multichip_mode

RSS Integration Register multichip mode

Type

bool

Default value

0x0

3.7.38 RandomNumberGenerator

Random Number Generator unit. This model is written in LISA+.

Iris and MTI instances for RandomNumberGenerator

This model has the following Iris instances:

Table 3-723: RandomNumberGenerator Iris instances

InstanceName	ComponentName
RandomNumberGenerator	RandomNumberGenerator
RandomNumberGenerator.pvbuslave	PVBusSlave

This model has the following MTI trace components:

Table 3-724: RandomNumberGenerator MTI instances

InstanceName	ComponentName
RandomNumberGenerator.pvbuslave	PVBusSlave

RandomNumberGenerator contains the following CADI targets:

- RandomNumberGenerator

Ports for RandomNumberGenerator

Table 3-725: Ports

Name	Protocol	Type	Description
pvbus_s	PVBus	Slave	Bus slave interface.
RNG_intr	Signal	Master	Interrupt output.

Parameters for RandomNumberGenerator

diagnostics

Diagnostics

Type

int

Default value

0x0

seed

Random number seed

Type

int

Default value

0x0

3.7.39 RealTimeLimiter

Real Time Limiter. This model is written in LISA+.

Iris and MTI instances for RealTimeLimiter

This model has the following Iris instances:

Table 3-726: RealTimeLimiter Iris instances

InstanceName	ComponentName
RealTimeLimiter	RealTimeLimiter
RealTimeLimiter.divider	ClockDivider

This model has the following MTI trace components:

Table 3-727: RealTimeLimiter MTI instances

InstanceName	ComponentName
RealTimeLimiter.divider	ClockDivider

RealTimeLimiter contains the following CADI targets:

- ClockDivider
- RealTimeLimiter

Ports for RealTimeLimiter

Table 3-728: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Clock input.

Parameters for RealTimeLimiter

ENABLE

Rate limit simulation

Type

bool

Default value

0x0

RELATIVE_SPEED

Rate limit to at most this percentage of real time (100: limit to wall clock rate)

Type

int

Default value

0x64

divider.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

3.7.40 RealtimeClockTimer

Host Time Based Timer Module for Generic Timers. This model is written in C++.

Ports for RealtimeClockTimer

Table 3-729: Ports

Name	Protocol	Type	Description
set_frequency	Value_64	Slave	-
timer_callback	TimerCallback64	Master	-
timer_control	TimerControl64	Slave	-

3.7.41 RemapDecoder

The component that provides support for dynamically remappable regions of memory. This model is written in LISA+.

Iris and MTI instances for RemapDecoder

This model has the following Iris instances:

Table 3-730: RemapDecoder Iris instances

InstanceName	ComponentName
RemapDecoder	RemapDecoder
RemapDecoder.bus_switch	TZSwitch
RemapDecoder.bus_switch.pvbus_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-731: RemapDecoder MTI instances

InstanceName	ComponentName
RemapDecoder.bus_switch.pvbus_mapper	PVBusMapper

RemapDecoder contains the following CADI targets:

- RemapDecoder
- TZSwitch

Ports for RemapDecoder

Table 3-732: Ports

Name	Protocol	Type	Description
control	TZSwitchControl	Broadcast	-
input	PVBus	Slave	Incoming bus transactions (connected straight to TZSwitch).
output_remap_clear	PVBus	Master	Outgoing bus transactions when remap is clear.

Name	Protocol	Type	Description
output_remap_set	PVBus	Master	Outgoing bus transactions when remap is set.
remap	StateSignal	Slave	Remapping control.

Parameters for RemapDecoder

bus_switch.normal

Normal Port

Type

int

Default value

0x2

bus_switch.secure

Secure Port

Type

int

Default value

0x1

3.7.42 RootKeyStorage

Trusted Root-Key Storage unit. This model is written in LISA+.

Iris and MTI instances for RootKeyStorage

This model has the following Iris instances:

Table 3-733: RootKeyStorage Iris instances

InstanceName	ComponentName
RootKeyStorage	RootKeyStorage
RootKeyStorage.pvbuslave	PVBusSlave

This model has the following MTI trace components:

Table 3-734: RootKeyStorage MTI instances

InstanceName	ComponentName
RootKeyStorage.pvbuslave	PVBusSlave

RootKeyStorage contains the following CADI targets:

- RootKeyStorage

Ports for RootKeyStorage

Table 3-735: Ports

Name	Protocol	Type	Description
pvbus_s	PVBus	Slave	Bus slave interface.

Parameters for RootKeyStorage

diagnostics

Diagnostics

Type

int

Default value

0x0

hw_unique_key

Hardware Unique Key (128-bit, 4 hex words)

Type

string

Default value

00000000 00000000 00000000 00000000

hw_unique_key_hex

Hardware Unique Key (128-bit, little-endian hex byte stream)

Type

string

Default value

private_key

Private Endorsement Key (256-bit, 8 hex words)

Type

string

Default value

00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000

private_key_hex

Private Key (256-bit, little-endian hex byte stream)

Type

string

Default value

public_key
Public Key (256-bit, 8 hex words)
Type
string
Default value
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

public_key_hex
Public Key (256-bit, little-endian hex byte stream)
Type
string
Default value

ss_key
Secret Symmetric Key (128-bit, 4 hex words)
Type
string
Default value
00000000 00000000 00000000 00000000

ss_key_hex
Secret Symmetric Key (128-bit, little-endian hex byte stream)
Type
string
Default value

version
Version of the model functionality. Valid values are r0 and r1.
Type
string
Default value
r1

3.7.43 SI_System_Ctrl_Regs

Safety Island System Control Registers. This model is written in C++.

Iris and MTI instances for SI_System_Ctrl_Regs

This model has the following Iris instances:

Table 3-736: SI_System_Ctrl_Regs Iris instances

InstanceName	ComponentName
SI_System_Ctrl_Regs	SI_System_Control_Registers
SI_System_Ctrl_Regs.PVBusSlave	PVBusSlave

This model has the following MTI trace components:

Table 3-737: SI_System_Ctrl_Regs MTI instances

InstanceName	ComponentName
SI_System_Ctrl_Regs.PVBusSlave	PVBusSlave

SI_System_Ctrl_Regs contains the following CADI targets:

- SI_System_Control_Registers

Ports for SI_System_Ctrl_Regs

Table 3-738: Ports

Name	Protocol	Type	Description
cpuhalt_m[7]	Signal	Master	-
pdbus_s	PVBus	Slave	-
reset_in	Signal	Slave	-
rvbar_cl0[1]	Value_64	Master	-
rvbar_cl1[2]	Value_64	Master	-
rvbar_cl2[4]	Value_64	Master	-

Parameters for SI_System_Ctrl_Regs

c10_c0_cfgrvbaraddr

CL0_CO_CFGRVBARADDR

Type

int

Default value

0x120000000

c11_c0_cfgrvbaraddr

CL1_CO_CFGRVBARADDR

Type

int

Default value

0x140000000

c11_c1_cfgrvbaraddr

CL1_C1_CFGRVBARADDR

Type

int

Default value

0x140002000

c12_c0_cfgrvbaraddr

CL2_C0_CFGRVBARADDR

Type

int

Default value

0x160000000

c12_c1_cfgrvbaraddr

CL2_C1_CFGRVBARADDR

Type

int

Default value

0x160002000

c12_c2_cfgrvbaraddr

CL2_C2_CFGRVBARADDR

Type

int

Default value

0x160004000

c12_c3_cfgrvbaraddr

CL2_C3_CFGRVBARADDR

Type

int

Default value

0x160008000

cpuhalt_reset

CPU HALT Reset value

Type

int

Default value

0x0

diagnostics

Diagnostics

Type

int

Default value

0x0

3.7.44 SMCF

System Monitoring Control Framework (SMCF). This model is written in C++.

Ports for SMCF

Table 3-739: Ports

Name	Protocol	Type	Description
mgi_clk_in	ClockSignal	Slave	Clock input
mgi_irq_out	Signal	Master	Interrupt signal output
mgi_pvbus_m	PVBus	Master	For DMA or memory mapped data write
mgi_reg_pvbus_s	PVBus	Slave	To access MGI register
mgi_reset_in	Signal	Slave	Reset signal input
mgi_tag_in[4]	Value	Slave	Tag value from external hardware
mgi_trigger_in	Signal	Slave	To trigger the start of a sample by external hardware
mgi_trigger_out	Signal	Master	Signal to external hardware to indicate that an event has occurred in an MGI
mli_amu_pvbus_m[32]	PVBus	Master	AMU manager port to read/write the AMU register value
mli_powerdown_in[32]	Signal	Slave	It gives info about AP is power OFF or not
mli_temperature_in[32]	ValueState	Slave	It will be connected to temperature sensor to fetch the temperature value

Parameters for SMCF

ALERT_NUM_CFG

Specifies the number of alerts present A value of 0 means that no alerts are present

Type

int

Default value

0x0

ALT_ADDR_CFG

Specifies where monitor data is read from, it is either: 0: The MGI_DATA<n> registers. 1: The address specified in MGI_RADDR0/1

Type

int

Default value

0x0

ALT_DELTA_CFG

Specifies the presence of alert rising and falling delta functions 0: The rising and falling delta functions are not present 1: The rising and falling delta functions are present

Type

int

Default value

0x0

DATA_PER_MON_CFG

Specifies the number of data values(DATA_PER_MON_CFG+1) generated from each monitor

Type

int

Default value

0x0

DEF_CFG_IRQ_MASK

Specifies the default value of the configuration interrupt event mask. Sets the reset value of MGI_IRQ_MASK.CFG_IRQ_MASK.

Type

int

Default value

0x1

DEF_CFG_TRIG_MASK

Specifies the default value of the configuration trigger event mask. Sets the reset value of MGI_TRG_MASK.CFG_TRIG_MASK.

Type

int

Default value

0x1

DEF_RADDR_CFG

Specifies the default alternate read address. This sets the default value for MGI_RADDR0/MGI_RADDR1

Type

int

Default value

0x0

DEF_WADDR_CFG

Specifies the default write address for the DMA interface. This sets the default value for MGI_WADDR0/MGI_WADDR1. Only required if DMA_IF_CFG = 1. This value must be 32-bit aligned

Type

int

Default value

0x0

DMA_IF_CFG

Specifies the presence of the DMA interface 0: DMA interface is not present 1: DMA interface is present

Type

int

Default value

0x0

END_COMPONENT

Based on this parameter value sampling value will be fetched from the respective model 0: Temperature sensor 1: Monitor unit (AMU) 2: Fake sensor/monitor

Type

int

Default value

0x0

FAKE_SENSOR_MAX_LIMIT

Maximum value limit for the fake sensor/monitor

Type

int

Default value

0xffff

FAKE_SENSOR_MIN_LIMIT

Minimum value limit for the fake sensor/monitor

Type

int

Default value

0x0

GRP_ID_CFG

Specifies a unique identifier for an MGI

Type

int

Default value

0x0

MGI_AIDR_RESET_VALUE

Reset value for MGI_AIDR register

Type

int

Default value

0x10

MGI_IIDR_RESET_VALUE

Reset value for MGI_IIDR register

Type

int

Default value

0x8d00043b

MLI_QUANTITY

Specifies the number of MLI

Type

int

Default value

0x1

MODE_LEN_CFG

Specifies the bit width of each MGI_MODE_REQ/STAT. The number of bits is MODE_LEN_CFG+1.

Type

int

Default value

0x1f

MODE_REG_CFG

Specifies the number of MGI_MODE_REQ/STAT pairs. A value of 0 means that no MGI_MODE_REQ/STAT pairs are present.

Type

int

Default value

0x0

MON_BASE_ADDRESS

AMU base address to which an offset is applied. It is used for configuration writes and data read bursts.

Type

int

Default value

0x0

MON_DATA_WIDTH_CFG

Specifies the bit width(MON_DATA_WIDTH_CFG+1) of each monitor data value

Type

int

Default value

0x1f

MON_DISCON_CFG

Specifies if each monitor supports being disconnected. For example, bit 0 represents monitor 0, and bit 3 represents monitor 3. 0b0: Monitor does not support being disconnected. It is reset to connected. 0b1: Monitor supports being disconnected. It is reset to disconnected.

Type

int

Default value

0x0

MON_NUM_CFG

Specifies the number of monitors(MON_NUM_CFG+1) in an MGI

Type

int

Default value

0x0

PACKED_CFG

Specifies if monitor data is packed 0: Data is not packed 1: Data is packed

Type

int

Default value

0x0

PER_TIMER_CFG

Specifies the presence of the periodic timer 0: The periodic timer is not present 1: The periodic timer is present

Type

int

Default value

0x0

SINGLE_MON_MODE_CFG

Specifies that each monitor is a single type and all monitors will have the same mode setting.

Type

int

Default value

0x0

SMP_DLY_LEN_CFG

Specifies the bit width of MGI_SMP_DLY. The number of bits in SMP_DLY_LEN_CFG. A value of 0 means this register is not present and the feature is not supported.

Type

int

Default value

0x0

TAG_IN_CFG

Specifies the presence of the tag input 0: The tag input is not present 1: The tag input is present

Type

int

Default value

0x1

TAG_LEN_CFG

Specifies the bit width of the tag input The tag bit width is TAG_LEN_CFG+1

Type

int

Default value

0x1f

TRIG_IN_CFG

Specifies the presence of the input trigger interface 0: The trigger in interface is not present 1: The trigger in interface is present

Type

int

Default value

0x1

TRIG_OUT_CFG

Specifies the presence of the output trigger interface 0: The trigger out interface is not present 1: The trigger out interface is present

Type

int

Default value

0x1

USER_DEF_CMD_CFG

Specifies if an MGI supports User-Defined commands. 0: User-Defined commands are not supported. 1: User-Defined commands are supported.

Type

int

Default value

0x0

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type

int

Default value

0x2

3.7.45 SMMUv3TestEngine

Test Engine used for testing SMMUv3. This model is written in C++.

Iris and MTI instances for SMMUv3TestEngine

This model has the following Iris instances:

Table 3-740: SMMUv3TestEngine Iris instances

InstanceName	ComponentName
SMMUv3TestEngine	SMMUv3TestEngine
SMMUv3TestEngine.register_file[0]	PVBusSlave

This model has the following MTI trace components:

Table 3-741: SMMUv3TestEngine MTI instances

InstanceName	ComponentName
SMMUv3TestEngine	SMMUv3TestEngine
SMMUv3TestEngine.register_file[0]	PVBusSlave

SMMUv3TestEngine contains the following CADI targets:

- SMMUv3TestEngine

Ports for SMMUv3TestEngine

Table 3-742: Ports

Name	Protocol	Type	Description
client_s	PCIDevice2ClientProtocol	Slave	-
clk_in	ClockSignal	Slave	-
identify	SMMUv3AEMIdentifyProtocol	Slave	-
pvbus_control_s	PVBus	Slave	-
pvbus_m[64]	PVBus	Master	-
reset_in	Signal	Slave	-

Parameters for SMMUv3TestEngine

bandwidth_per_transaction_in_bytes_per_tick

The bandwidth of the device for each in-flight transaction, in bytes/tick of clk_in. This is only a rough guess. If you are uninterested in trying to run cores and the engine simultaneously then set this to a large number.

Type

int

Default value

0x64

max_number_of_inflight_transactions

The maximum number of in-flight transactions allowed.

Type

int

Default value

0xa

output_attribute_transform

How to pack the stream identification information into the transaction attributes. This is: - * <empty> or "default" * "pcie" the de-facto standard for the PCIe subsystem in FastModels

The <empty> or "default" is equivalent to:- ExtendedID[63]=nSEC_SID, ExtendedID[55:24]=StreamID, ExtendedID[20]=nSSV, ExtendedID[19:0]=SubstreamID

The "pcie" option is equivalent to:- ExtendedID[63]=SEC_SID, ExtendedID[62]=SSV, ExtendedID[51:32]=SubstreamID, ExtendedID[31:0]=StreamID

Type

string

Default value

seed

The seed to use for initialising the random number generators.

Type

int

Default value

0x12345678

3.7.46 STLBusGasket

STLBusGasket allows a debugger or emulated T32 code to force the results of system-register reads by writing an address to the ADDR register then 32-bit values to the VALUE register, which are placed in a fifo associated with that address. A PVBus transaction into pvbus_in goes unchanged to pvbus_out, unless its address matches that associated with a non-empty fifo, in which case: writes are ignored, non-word reads abort, and word reads take values from that fifo. This model is written in LISA+.

Iris and MTI instances for STLBusGasket

This model has the following Iris instances:

Table 3-743: STLBusGasket Iris instances

InstanceName	ComponentName
STLBusGasket	STLBusGasket
STLBusGasket.busmapper	PVBusMapper

This model has the following MTI trace components:

Table 3-744: STLBusGasket MTI instances

InstanceName	ComponentName
STLBusGasket.busmapper	PVBusMapper

STLBusGasket contains the following CADI targets:

- STLBusGasket

Ports for STLBUSGasket

Table 3-745: Ports

Name	Protocol	Type	Description
pdbus_in	PVBus	Slave	-
pdbus_out	PVBus	Master	-

Parameters for STLBUSGasket

function

Function: 0-none, 1-STL value-forcing

Type

int

Default value

0x0

reg_base

Base Address of STL control regs (ADDR,VAL at offsets 0,4)

Type

int

Default value

0xe001e820

verbose

Verbosity : 0-none, 1-some

Type

int

Default value

0x0

3.7.47 SerialCrossover

Implement an equivalent to a null-modem cable, swapping over serial transmit and receive signals. This model is written in C++.

About SerialCrossover

This component implements two SerialData slave ports and can connect two SerialData master ports, such as from PL011_Uart components. Data received on one port is buffered in a FIFO until it is read from the other port. Signals received on one port are latched and available to be read by the other port.

Ports for SerialCrossover

Table 3-746: Ports

Name	Protocol	Type	Description
port_a	SerialData	Slave	Slave port for connecting to a SerialData master.
port_b	SerialData	Slave	Slave port for connecting to a SerialData master.

3.7.48 TelnetTerminal

A host interface onto a serial port: exposes the two way serial data channel over a TCP/IP interface, and automatically opens a telnet application connected to the network socket, unless a user application connects first. This model is written in C++.

Iris and MTI instances for TelnetTerminal

This model has the following Iris instances:

Table 3-747: TelnetTerminal Iris instances

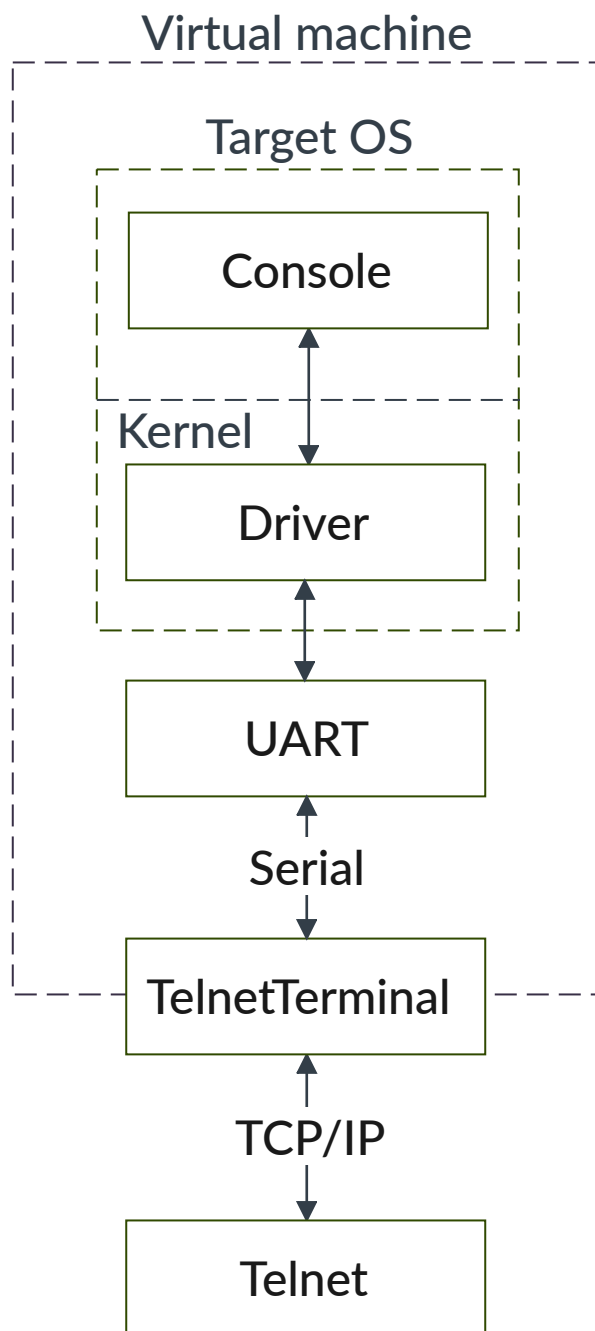
InstanceName	ComponentName
TelnetTerminal	TelnetTerminal

TelnetTerminal contains the following CADI targets:

- TelnetTerminal

Using TelnetTerminal

The following figure shows a block diagram of one possible relationship between the target and host through the TelnetTerminal component. The TelnetTerminal block is what you configure when you define Terminal component parameters. The Virtual Machine is an FVP.

Figure 3-5: Terminal block diagram

On the target side, the console process that is invoked by your target OS relies on a suitable driver being present. Such drivers are normally part of the OS kernel. The driver passes serial data

through a UART, for example [3.10.62 PL011_Uart](#) on page 3949. The data is forwarded to the TelnetTerminal component. When the simulation is started and the TelnetTerminal component is enabled, the component opens a server (listening) socket on a TCP/IP port. This is port 5000 by default. This port can be connected to by, for example, a Telnet process on the host.

When data becomes available on the network socket, the TelnetTerminal component buffers the data, which can then be read from SerialData.

If there is no connection to the network socket when the first data access is made, and the `start_telnet` parameter is true, a host Telnet session is started automatically. Prior to this first access, you can connect a client of your choice to the network socket.

If the connection between the TelnetTerminal component and the client is broken at any time, for example by closing a client Telnet session, the port is re-opened on the host, permitting you to make another client connection. This could have a different port number if the original one is no longer available.

The port number of a particular TelnetTerminal instance can be defined when your model system starts. The actual value of the port used by each TelnetTerminal is declared when it starts or restarts, and might not be the value that you specified if the port is already in use. If you are using Model Shell, the port numbers are displayed in the host window in which you started the model.

Microsoft Windows 10 disables the Telnet client by default. Follow these steps to enable it:



Note

1. Select **Start > Settings**.
 2. In the search box, type **Turn Windows features on or off**. The **Windows Features** dialog opens.
 3. Select the **Telnet Client** check box and click **OK**. The installation might take several minutes to complete.
-

TelnetTerminal parameters

To set the parameters, the syntax to use in a configuration file or on the command line is:

```
motherboard.terminal_x.parameter=value
```

where *x* is the terminal identifier and can be 0, 1, 2, or 3.

You can start the TelnetTerminal component in either of the following modes, depending on the `mode` parameter:

telnet

In Telnet mode, the terminal component supports a subset of the RFC 854 protocol. This means that the terminal participates in negotiations between the host and client concerning what is and is not supported, but there is no flow control.

raw

In raw mode the byte stream passes unmodified between the host and the target. The terminal does not participate in initial capability negotiations between the host and client. Instead it acts as a TCP/IP port. You can use this feature to directly connect to your target through the TelnetTerminal component. This permits a debugger connection, for example, to connect a gdb client to a gdbserver running on the target operating system.

The `terminal_command` parameter specifies the command line used to launch a terminal application and connect to the opened TCP port. The TelnetTerminal component replaces the keywords `%port` and `%title`, if specified, with the opened port number and component name, respectively. After replacing `%port` and `%title`, the command line is executed verbatim.

An empty string, which is the default, launches `xterm` on Linux or `telnet.exe` on Windows.



If you specify a non-empty string, it must include `%port`, but `%title` is optional.

For example:

```
fvp_mps2.telnetterminal0.terminal_command="putty.exe -telnet localhost %port"
```

Ports for TelnetTerminal

Table 3-748: Ports

Name	Protocol	Type	Description
serial	SerialData	Slave	Slave port for connecting to a SerialData master.

Parameters for TelnetTerminal

mode

Terminal initialisation mode

Type

string

Default value

telnet

quiet

Avoid output on stdout/stderr

Type

bool

Default value

0x0

start_port

Telnet TCP Port Number

Type

int

Default value

0x1388

start_telnet

Start telnet if nothing connected

Type

bool

Default value

0x1

terminal_command

Commandline to launch a terminal application and connect to the opened TCP port. Keywords %port and %title will be replaced with the opened port number and component name respectively. An empty string (default behaviour) will launch xterm (Linux) or telnet.exe (Windows)

Type

string

Default value

3.7.49 UnusedPrimeCell

A dummy component. It can be used to represent any unimplemented PrimeCell components. This model is written in LISA+.

Iris and MTI instances for UnusedPrimeCell

This model has the following Iris instances:

Table 3-749: UnusedPrimeCell Iris instances

InstanceName	ComponentName
UnusedPrimeCell	UnusedPrimeCell
UnusedPrimeCell.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-750: UnusedPrimeCell MTI instances

InstanceName	ComponentName
UnusedPrimeCell.busslave	PVBusSlave

UnusedPrimeCell contains the following CADI targets:

- UnusedPrimeCell

Ports for UnusedPrimeCell

Table 3-751: Ports

Name	Protocol	Type	Description
pvbust	PVBus	Slave	Bus slave interface.

3.7.50 VirtioBlockDevice

virtio block device. This model is written in C++.

Iris and MTI instances for VirtioBlockDevice

This model has the following Iris instances:

Table 3-752: VirtioBlockDevice Iris instances

InstanceName	ComponentName
VirtioBlockDevice	VirtioBlockDevice
VirtioBlockDevice.register_slave	PVBusSlave
VirtioBlockDevice.virtio_master	PVBusMaster

This model has the following MTI trace components:

Table 3-753: VirtioBlockDevice MTI instances

InstanceName	ComponentName
VirtioBlockDevice	VirtioBlockDevice
VirtioBlockDevice.register_slave	PVBusSlave
VirtioBlockDevice.virtio_master	PVBusMaster

VirtioBlockDevice contains the following CADI targets:

- VirtioBlockDevice

About VirtioBlockDevice

VirtioBlockDevice implements a block device that can be accessed from the simulated OS if it has an appropriate driver. Similarly to the VirtioP9Device, this component is targeted primarily at Linux, which has a built-in virtio block driver. VirtioBlockDevice allows you to use a file on the host that you specify using the `image_path` parameter, as a hard drive in the simulated OS.

VirtioBlockDevice supports the legacy OASIS virtio specification.

Unlike the VirtioP9Device, you should not need to carry out any special setup to use VirtioBlockDevice on VE or Base platforms, because it is usually already included in the device

trees. Set the `image_path` parameter to point to your image, and then on Linux it is available as a block device, usually `/dev/vda`, which you then use like any other hard drive.

Ports for VirtioBlockDevice

Table 3-754: Ports

Name	Protocol	Type	Description
<code>intr</code>	Signal	Master	Virtio device sets interrupt to signal completion.
<code>pvbus</code>	PVBus	Slave	Virtio MMIO control/config/status registers.
<code>virtio_m</code>	PVBus	Master	Virtio device performs DMA accesses via master.

Parameters for VirtioBlockDevice

`image_path`

image file path

Type

string

Default value

`quiet`

Don't print warnings on malformed commands/descriptors

Type

bool

Default value

0x0

`read_only`

Only allow device to be read

Type

bool

Default value

0x0

`secure_accesses`

Make device generate transactions with NS=0

Type

bool

Default value

0x0

`transaction_attributes`

Transaction attributes used by device. 0x0 - inner-shared real access. 0x1 - outer-shared real access. 0x2 - outer-shared debug access.

Type

int

Default value

0x0

3.7.51 VirtioBlockDeviceMMIO

Virtio v1.0 Block device over MMIO transport. This model is written in C++.

Iris and MTI instances for VirtioBlockDeviceMMIO

This model has the following Iris instances:

Table 3-755: VirtioBlockDeviceMMIO Iris instances

InstanceName	ComponentName
VirtioBlockDeviceMMIO	VirtioBlockMMIO
VirtioBlockDeviceMMIO.dma_master	PVBusMaster

This model has the following MTI trace components:

Table 3-756: VirtioBlockDeviceMMIO MTI instances

InstanceName	ComponentName
VirtioBlockDeviceMMIO.dma_master	PVBusMaster

VirtioBlockDeviceMMIO contains the following CADI targets:

- VirtioBlockMMIO

About VirtioBlockDeviceMMIO

VirtioBlockDeviceMMIO supports both the legacy and v1.0 OASIS virtio specifications.



VirtioBlockDeviceMMIO is an evolution of [3.7.50 VirtioBlockDevice](#) on page 2998, which is also MMIO-based and has the same ports, but only supports the legacy OASIS virtio specification.

Ports for VirtioBlockDeviceMMIO

Table 3-757: Ports

Name	Protocol	Type	Description
intr	Signal	Master	Virtio device sets interrupt to signal completion.
pvbus	PVBus	Slave	Virtio MMIO control/config/status registers.
virtio_m	PVBus	Master	Virtio device performs DMA accesses via master.

Parameters for VirtioBlockDeviceMMIO

enabled

Enable or disable device. If disabled, device can be accessed, but will not be activated

Type

bool

Default value

0x0

image_path

Image file path

Type

string

Default value

quiet

Don't print info or warnings (e.g. on malformed commands/descriptors)

Type

bool

Default value

0x0

read_only

Only allow device to be read. If that parameter is set to false and the image file cannot be opened in RW mode, the model will try to work around it by opening the file in RO mode

Type

bool

Default value

0x0

secure_accesses

Make device generate transactions with NS=0

Type

bool

Default value

0x0

transaction_attributes

Transaction attributes used by device. 0x0 - inner-shared real access. 0x1 - outer-shared real access. 0x2 - outer-shared debug access.

Type

int

Default value

0x0

transport

Choose legacy or modern virtio transport, if not specified, modern transport is used

Type

string

Default value

modern

3.7.52 VirtioNetMMIO

Virtio net device over MMIO transport. This model is written in C++.

Changes in 11.26.8

Parameters added:

- checksum

Iris and MTI instances for VirtioNetMMIO

This model has the following Iris instances:

Table 3-758: VirtioNetMMIO Iris instances

InstanceName	ComponentName
VirtioNetMMIO	VirtioNetMMIO
VirtioNetMMIO.dma_master	PVBusMaster
VirtioNetMMIO.hostbridge	HostBridge

This model has the following MTI trace components:

Table 3-759: VirtioNetMMIO MTI instances

InstanceName	ComponentName
VirtioNetMMIO.dma_master	PVBusMaster

VirtioNetMMIO contains the following CADI targets:

- HostBridge
- VirtioNetMMIO

About VirtioNetMMIO

This is a model of a virtual Ethernet virtio device over MMIO transport, supporting both the legacy and v1.0 OASIS virtio specifications. It provides much better network performance than the SMSC_91C111 component, because it features host-assisted network acceleration. This means

that it can offload packet processing operations from the simulated OS on the target, to the host side. These operations include:

- Checksum computation
- TX packet segmentation
- RX packet combination

If the target simulated Linux or Linux-derived OS has an appropriate virtio net driver, Arm recommends you use VirtioNetMMIO instead of SMSC_91C111.

Unlike SMSC_91C111, which must work with an external HostBridge component, VirtioNetMMIO has a built-in HostBridge sub-component. The parameters to control the HostBridge are described in the VirtioNetMMIO parameters table, with the `hostbridge` parameter sub-namespace.

To enable tracing of user-mode networking, which can help to debug networking issues, set the `FASTSIM_USERNET_DUMP` environment variable to any or all of the following values:

```
arpin, arpout, udpin, udpout, etherin, etherout, ipv4in, ipv4out, ipv4fragin, ipv4fragout, tcpin, tcpout
```

Take the following steps to set up this component in a virtual platform:

- Use a version of Linux that contains a virtio network driver.
- Add the following option to the Linux kernel configuration:

```
CONFIG_VIRTIO_NET=y
```

- Update the device tree to include the `virtioNetMMIO` component, or specify it on the kernel command line, for example

```
virtio_mmio.device=0x10000@0x1c150000:76
```

The address range for both VE and Base platforms is `0x1C150000-0x1C15FFFF`. The interrupt number is 44, or IRQ 76, for both VE and Base platforms.

- Select the hostbridge that you want to use to communicate with the host in the model:

```
virtio_net.hostbridge.userNetworking=true/false (User mode or TAP/TUN networking)
```

- Configure the networking environment, as described in the related tasks at the end of this topic.

Example entries for DTS files:

- Add this entry next to the corresponding `virtio_block` or `virtio_p9` entry:

```
virtio_net@0150000 {
    compatible = "virtio,mmio";
    reg = <0x150000 0x1000>;
    interrupts = <0x2c>;
};
```

- Add this entry to the interrupt map:

```
<0 0 44 &gic 0 44 4>;
```

See also:

- [1.11.3 Configuring the networking environment for Microsoft Windows](#) on page 47
- [1.11.8 Configuring the networking environment for Linux](#) on page 53

Ports for VirtioNetMMIO

Table 3-760: Ports

Name	Protocol	Type	Description
intr	Signal	Master	Virtio device sets interrupt to signal completion.
pvbus	PVBus	Slave	Virtio MMIO control/config/status registers.
virtio_m	PVBus	Master	Virtio device performs DMA accesses via master.

Parameters for VirtioNetMMIO

checksum

For checksum-offloaded packets, if 'tx' is specified, outgoings will be checksummed by VirtioNet device; 'rx' is specified for incomings; 'all' for both

Type

string

Default value

enabled

Enable or disable device. If disabled, device can be accessed, but will not be activated

Type

bool

Default value

0x1

hostbridge.interfaceName

Host Interface

Type

string

Default value

hostbridge.userNetOptions

Control options for UserNet TCP/IP (for internal use only, please do not use)

Type

string

Default value**hostbridge.userNetPorts**

Listening ports to expose in user-mode networking

Type

string

Default value**hostbridge.userNetSubnet**

Virtual subnet for user-mode networking

Type

string

Default value

172.20.51.0/24

hostbridge.userNetworking

Enable user-mode networking

Type

bool

Default value

0x1

mac_address

Device MAC address, if not specified, a random MAC address is generated

Type

string

Default value**offload**

Offload TCP/UDP segmentation/receiving operations to host

Type

string

Default value

gso, gro

secure_accesses

Make device generate transactions with NS=0

Type

bool

Default value

0x0

transport

Choose legacy or modern virtio transport, if not specified, modern transport is used

Type

string

Default value

modern

3.7.53 VirtioP9Device

virtio P9 server. This model is written in C++.

Iris and MTI instances for VirtioP9Device

This model has the following Iris instances:

Table 3-761: VirtioP9Device Iris instances

InstanceName	ComponentName
VirtioP9Device	VirtioP9Device
VirtioP9Device.mmio_slave	PVBusSlave
VirtioP9Device.virtio_master	PVBusMaster

This model has the following MTI trace components:

Table 3-762: VirtioP9Device MTI instances

InstanceName	ComponentName
VirtioP9Device	VirtioP9Device
VirtioP9Device.mmio_slave	PVBusSlave
VirtioP9Device.virtio_master	PVBusMaster

VirtioP9Device contains the following CADI targets:

- VirtioP9Device

About VirtioP9Device

This component implements a subset of the Plan 9 file protocol over a virtio transport. It enables accessing a directory on the host's filesystem within Linux, or another operating system that implements the protocol, running on a platform model.

It supports the legacy OASIS virtio specification.

It implements a subset of the Linux 9P2000.L protocol. It has the following limitations:

- You can mount only one host directory per instance of the component.

- It supports a subset of 9P2000.L message types: Tversion, Tlopen, Tlcreate, Tgetattr, Tsetattr, Treaddir, Tmkdir, Tattach, Twalk, Tread, Twrite, Tclunk, Tremove, Trename. On Linux hosts, it also supports: Treadlink, Tsymlink.
- On Windows hosts, it ignores Unix permissions when writing files.
- On Windows hosts, it performs a simple mapping from Windows to Unix permissions when reading.
- On Windows hosts, symbolic links appear as regular files.
- On Windows hosts, it does not perform writing, deleting, or renaming operations on a file that another process has open.

Setting up the VirtioP9Device component

Take the following steps to set up this component:

- Use a version of Linux that supports v9fs over virtio and virtio-mmio devices.
- Update the device tree to include the VirtioP9Device component, or specify it on the kernel command-line, as shown below. The address range for both VE and Base platforms is 0x1C140000-0x1C14FFFF. The interrupt number is 43, or IRQ 75, for both VE and Base platforms.
- Set the following parameter to the directory on the host that you want to mount in the model:

VE:

```
motherboard.virtiop9device.root_path
```

Base:

```
bp.virtiop9device.root_path
```

- On Linux, mount the host directory by using the following command in the model:

```
$ mount -t 9p -o trans=virtio,version=9p2000.L FM <mount point>
```

Example kernel command-line argument:

```
virtio_mmio.device=0x10000@0x1c140000:75
```

Example entry for DTS files, to add next to the corresponding `virtio_block` entry:

```
virtio_p9@0140000 {
    compatible = "virtio,mmio";
    reg = <0x0 0x1c140000 0x0 0x1000>;
    interrupts = <0x0 0x2b 0x4>;
};
```

Ports for VirtioP9Device

Table 3-763: Ports

Name	Protocol	Type	Description
intr	Signal	Master	Virtio device sets interrupt to signal completion.
pvbus	PVBus	Slave	Virtio MMIO control/config/status registers.

Name	Protocol	Type	Description
virtio_m	PVBus	Master	Virtio device performs DMA accesses via master.

Parameters for VirtioP9Device

- mount_tag**

mount tag

Type

string

Default value

FM
- quiet**

Don't print warnings on malformed commands/descriptors

Type

bool

Default value

0x0
- root_path**

root directory path

Type

string

Default value
- secure_accesses**

Make device generate transactions with NS=0

Type

bool

Default value

0x0

3.7.54 VirtioPCIBlockDevice

virtio block device with PCI transport. This model is written in C++.

Iris and MTI instances for VirtioPCIBlockDevice

This model has the following Iris instances:

Table 3-764: VirtioPCIBlockDevice Iris instances

InstanceName	ComponentName
VirtioPCIBlockDevice	VirtioPCIBlockDevice
VirtioPCIBlockDevice.register_slave	PVBusSlave
VirtioPCIBlockDevice.virtio_master	PVBusMaster

This model has the following MTI trace components:

Table 3-765: VirtioPCIBlockDevice MTI instances

InstanceName	ComponentName
VirtioPCIBlockDevice	VirtioPCIBlockDevice
VirtioPCIBlockDevice.register_slave	PVBusSlave
VirtioPCIBlockDevice.virtio_master	PVBusMaster

VirtioPCIBlockDevice contains the following CADI targets:

- VirtioPCIBlockDevice

About VirtioPCIBlockDevice

VirtioPCIBlockDevice is similar to VirtioBlockDevice, except it is PCI-based instead of MMIO-based. It supports the legacy OASIS virtio specification.

Ports for VirtioPCIBlockDevice

Table 3-766: Ports

Name	Protocol	Type	Description
client_s	PCIDevice2ClientProtocol	Slave	Interrupts for MSI-X table entries.
intr	Signal	Master	Virtio device sets interrupt to signal completion.
pvbus	PVBus	Slave	Virtio pci/control/config/status registers.
virtio_m	PVBus	Master	Virtio device performs DMA accesses via master.

Parameters for VirtioPCIBlockDevice

image_path

image file path

Type

string

Default value

quiet

Don't print warnings on malformed commands/descriptors

Type

bool

Default value

0x0

read_only

Only allow device to be read

Type

bool

Default value

0x0

secure_accesses

Make device generate transactions with NS=0

Type

bool

Default value

0x0

transaction_attributes

Transaction attributes used by device. 0x0 - inner-shared real access. 0x1 - outer-shared real access. 0x2 - outer-shared debug access.

Type

int

Default value

0x0

3.7.55 VirtioRNG

virtio rng - Entropy device. This model is written in C++.

Changes in 11.26.8

Parameters added:

- secure_accesses
- transport

Iris and MTI instances for VirtioRNG

This model has the following Iris instances:

Table 3-767: VirtioRNG Iris instances

InstanceName	ComponentName
VirtioRNG	VirtioEntropyMMIO
VirtioRNG.dma_master	PVBusMaster

This model has the following MTI trace components:

Table 3-768: VirtioRNG MTI instances

InstanceName	ComponentName
VirtioRNG.dma_master	PVBusMaster

VirtioRNG contains the following CADI targets:

- VirtioEntropyMMIO

About VirtioRNG

VirtioRNG models a virtio entropy device as defined in the virtio 1.0 specification at <http://docs.oasis-open.org/virtio/virtio/v1.0/virtio-v1.0.html>. A virtual platform might need to integrate a VirtioRNG component to generate random numbers when:

- Linux or Android needs to generate kernel entropy. Hardware might do this using a timer, but this is not possible in the model because timers are not updated quickly enough.
- Security features are required, such as ssh.

Integrate VirtioRNG into a platform

Integrate the VirtioRNG component by instantiating it in your board's LISA file and connecting it to the SoC virtio master bus and interrupt signal as follows:

```
// Instantiate components
composition {
...
    virtio_rng : VirtioRNG();
...
}

connection {
...
    // Find a suitable address space and connect it to the SoC's virtio_m bus
    busdecoder.pvbus_m_range[0x001C190000..0x001C19ffff] => virtio_rng.pvbus;
    virtio_rng.virtio_m => self.virtio_m;

    // Connect the IRQ to the GIC IRQ
    virtio_rng.intr => gic400.irqs[101];
...
}
```

To configure Linux or Android for VirtioRNG, use the following build parameters:

Table 3-769: Linux and Android build parameters

Linux	Android
CONFIG_VIRTIO_MMIO=y	--enable CONFIG_VIRTIO_MMIO
CONFIG_HW_RANDOM=y	--enable CONFIG_HW_RANDOM
CONFIG_HW_RANDOM_VIRTIO=y	--enable CONFIG_HW_RANDOM_VIRTIO

Use the following device tree parameters:

```
virtio_rng@1c190000 {
```

```
compatible = "virtio,mmio";
reg = <0x0 0x1c190000 0x0 0x200>;
interrupts = <GIC_SPI 101 IRQ_TYPE_LEVEL_HIGH>;
};
```

Configure VirtioRNG using model parameters, for example:

```
-C "board.virtio_rng.enabled=1" \
-C "board.virtio_rng.seed=0" \
-C "board.virtio_rng.generator=2" \
-C "board.virtio_rng.diagnostics=4" \ # Optional
```

Use the following guest command line to test the integration:

```
// Generate random numbers
#console/> cat /dev/hwrng
```

Ports for VirtioRNG

Table 3-770: Ports

Name	Protocol	Type	Description
intr	Signal	Master	Virtio device sets interrupt to signal completion.
pvbus	PVBus	Slave	Virtio control/config/status registers.
virtio_m	PVBus	Master	Virtio device performs DMA accesses via master.

Parameters for VirtioRNG

- diagnostics**

Prints debug information: 0 = disabled; 1 = generated seed and device; 4 = generated seed, device and generated numbers

Type

int

Default value

0x0
- enabled**

Enable or disable device. If disabled, device can be accessed, but will not be activated

Type

bool

Default value

0x0
- generator**

User-defined generator: 0 = xorshiftstar; 1 = rand48; 2 = mersenne

Type

int

Default value

0x0

secure_accesses

Make device generate transactions with NS=0

Type

bool

Default value

0x0

seed

User-defined seed: 0 = uses a random seed; > 0 = user-defined fixed seed value

Type

int

Default value

0x0

transport

Choose legacy or modern virtio transport, if not specified, modern transport is used

Type

string

Default value

modern

3.7.56 VirtualEthernetCrossover

Ethernet Crossover Cable. This model is written in LISA+.

Iris and MTI instances for VirtualEthernetCrossover

This model has the following Iris instances:

Table 3-771: VirtualEthernetCrossover Iris instances

InstanceName	ComponentName
VirtualEthernetCrossover	VirtualEthernetCrossover

This model has the following MTI trace components:

Table 3-772: VirtualEthernetCrossover MTI instances

InstanceName	ComponentName
VirtualEthernetCrossover	VirtualEthernetCrossover

VirtualEthernetCrossover contains the following CADI targets:

- VirtualEthernetCrossover

About VirtualEthernetCrossover

This component implements two VirtualEthernet slave ports and enables you to connect two VirtualEthernet master ports. It forwards data received on one port to the other port without delay.

Ports for VirtualEthernetCrossover

Table 3-773: Ports

Name	Protocol	Type	Description
deva	VirtualEthernet	Slave	Slave port for connecting to a VirtualEthernet master.
devb	VirtualEthernet	Slave	Slave port for connecting to a VirtualEthernet master.

3.7.57 VirtualEthernetHub3

3 Port Ethernet Hub. This model is written in LISA+.

Iris and MTI instances for VirtualEthernetHub3

This model has the following Iris instances:

Table 3-774: VirtualEthernetHub3 Iris instances

InstanceName	ComponentName
VirtualEthernetHub3	VirtualEthernetHub3

VirtualEthernetHub3 contains the following CADI targets:

- VirtualEthernetHub3

Ports for VirtualEthernetHub3

Table 3-775: Ports

Name	Protocol	Type	Description
deva	VirtualEthernet	Slave	Slave port for connecting to a VirtualEthernet master.
devb	VirtualEthernet	Slave	Slave port for connecting to a VirtualEthernet master.
devc	VirtualEthernet	Slave	Slave port for connecting to a VirtualEthernet master.

3.7.58 VisEventRecorder

Event recorder component for visualisation component (allows to playback and record interactive GUI sessions). This model is written in LISA+.

Iris and MTI instances for VisEventRecorder

This model has the following Iris instances:

Table 3-776: VisEventRecorder Iris instances

InstanceName	ComponentName
VisEventRecorder	VisEventRecorder
VisEventRecorder.playbackDivider	ClockDivider
VisEventRecorder.playbackTimer	ClockTimerThread
VisEventRecorder.playbackTimer.timer	ClockTimerThread64
VisEventRecorder.playbackTimer.timer.thread	SchedulerThread
VisEventRecorder.playbackTimer.timer.thread_event	SchedulerThreadEvent
VisEventRecorder.recordingDivider	ClockDivider

This model has the following MTI trace components:

Table 3-777: VisEventRecorder MTI instances

InstanceName	ComponentName
VisEventRecorder.playbackDivider	ClockDivider
VisEventRecorder.recordingDivider	ClockDivider

VisEventRecorder contains the following CADI targets:

- ClockDivider
- ClockTimerThread
- ClockTimerThread64
- SchedulerThread
- SchedulerThreadEvent
- VisEventRecorder

Ports for VisEventRecorder

Table 3-778: Ports

Name	Protocol	Type	Description
control	VisEventRecorderProtocol	Slave	The visualisation component controls the recorder through this port.
ticks	InstructionCount	Slave	Allow VisEventRecorder to get tick count from a core.

Parameters for VisEventRecorder

checkInstructionCount

check instruction count in recording file against actual instruction count during playback

Type

bool

Default value

0x1

playbackDivider.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

playbackFileName

playback filename (empty string disables playback)

Type

string

Default value**recordingDivider.div**

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

recordingFileName

recording filename (empty string disables recording)

Type

string

Default value**recordingTimeBase**

timebase in 1/s (relative to the master clock (e.g. 100000000 means 10 nanoseconds resolution simulated time for a 1Hz master clock)) to be used for recording (higher values -> higher time resolution, playback time base is always taken from the playback file)

Type

int

Default value

0x5f5e100

verbose

enable verbose messages (1=normal, 2=even more)

Type

int

Default value

0x0

3.7.59 Visualisation_sdl2_cpp

Display window for VE using sdl2 Visualisation library. This model is written in C++.

Iris and MTI instances for Visualisation_sdl2_cpp

This model has the following Iris instances:

Table 3-779: Visualisation_sdl2_cpp Iris instances

InstanceName	ComponentName
Visualisation_sdl2_cpp	Visualisation_sdl2
Visualisation_sdl2_cpp.VisEventRecorder_cpp	VisEventRecorder
Visualisation_sdl2_cpp.VisEventRecorder_cpp.ClockDivider	ClockDivider
Visualisation_sdl2_cpp.VisEventRecorder_cpp.ClockTimerThread	ClockTimerThread
Visualisation_sdl2_cpp.VisEventRecorder_cpp.ClockTimerThread.ClockTimerThread64	ClockTimerThread64
Visualisation_sdl2_cpp.VisEventRecorder_cpp.ClockTimerThread.ClockTimerThread64.SchedulerThread	SchedulerThread
Visualisation_sdl2_cpp.VisEventRecorder_cpp.ClockTimerThread.ClockTimerThread64.SchedulerThreadEvent	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-780: Visualisation_sdl2_cpp MTI instances

InstanceName	ComponentName
Visualisation_sdl2_cpp.VisEventRecorder_cpp.ClockDivider	ClockDivider

Visualisation_sdl2_cpp contains the following CADI targets:

- Visualisation_sdl2

Ports for Visualisation_sdl2_cpp

Table 3-781: Ports

Name	Protocol	Type	Description
c0_core_freq[16]	ValueState	Slave	-
c10_core_freq[16]	ValueState	Slave	-
c11_core_freq[16]	ValueState	Slave	-
c12_core_freq[16]	ValueState	Slave	-
c13_core_freq[16]	ValueState	Slave	-
c14_core_freq[16]	ValueState	Slave	-
c15_core_freq[16]	ValueState	Slave	-
c1_core_freq[16]	ValueState	Slave	-
c2_core_freq[16]	ValueState	Slave	-
c3_core_freq[16]	ValueState	Slave	-
c4_core_freq[16]	ValueState	Slave	-

Name	Protocol	Type	Description
c5_core_freq[16]	ValueState	Slave	-
c6_core_freq[16]	ValueState	Slave	-
c7_core_freq[16]	ValueState	Slave	-
c8_core_freq[16]	ValueState	Slave	-
c9_core_freq[16]	ValueState	Slave	-
clock_50Hz	ClockSignal	Slave	-
cluster0_ticks[16]	InstructionCount	Slave	-
cluster10_ticks[16]	InstructionCount	Slave	-
cluster11_ticks[16]	InstructionCount	Slave	-
cluster12_ticks[16]	InstructionCount	Slave	-
cluster13_ticks[16]	InstructionCount	Slave	-
cluster14_ticks[16]	InstructionCount	Slave	-
cluster15_ticks[16]	InstructionCount	Slave	-
cluster1_ticks[16]	InstructionCount	Slave	-
cluster2_ticks[16]	InstructionCount	Slave	-
cluster3_ticks[16]	InstructionCount	Slave	-
cluster4_ticks[16]	InstructionCount	Slave	-
cluster5_ticks[16]	InstructionCount	Slave	-
cluster6_ticks[16]	InstructionCount	Slave	-
cluster7_ticks[16]	InstructionCount	Slave	-
cluster8_ticks[16]	InstructionCount	Slave	-
cluster9_ticks[16]	InstructionCount	Slave	-
cluster_freq[16]	ValueState	Slave	-
keyboard	KeyboardStatus	Master	-
lcd	LCD	Slave	-
lcd_layout	LCDLayoutInfo	Master	-
mcp_freq	ValueState	Slave	-
mcp_ticks	InstructionCount	Slave	-
mouse	MouseStatus	Master	-
poreset	Signal	Master	-
scp_freq	ValueState	Slave	-
scp_ticks	InstructionCount	Slave	-
sys_temperature[16]	ValueState	Slave	-
touch_screen	MouseStatus	Master	-

Parameters for Visualisation_sdl2_cpp

cluster0_name

Cluster0 name

Type

string

Default value**cluster10_name**

Cluster10 name

Type

string

Default value**cluster11_name**

Cluster11 name

Type

string

Default value**cluster12_name**

Cluster12 name

Type

string

Default value**cluster13_name**

Cluster13 name

Type

string

Default value**cluster14_name**

Cluster14 name

Type

string

Default value**cluster15_name**

Cluster15 name

Type

string

Default value

cluster1_name

Cluster1 name

Type

string

Default value**cluster2_name**

Cluster2 name

Type

string

Default value**cluster3_name**

Cluster3 name

Type

string

Default value**cluster4_name**

Cluster4 name

Type

string

Default value**cluster5_name**

Cluster5 name

Type

string

Default value**cluster6_name**

Cluster6 name

Type

string

Default value**cluster7_name**

Cluster7 name

Type

string

Default value

cluster8_name

Cluster8 name

Type

string

Default value

cluster9_name

Cluster9 name

Type

string

Default value

css_spec

Platform specification displayed in window title

Type

string

Default value

Columbus mid

diagnostics

Diagnostics

Type

int

Default value

0x0

disable_visualisation

Enable/disable visualisation

Type

bool

Default value

0x0

display_object

Display objects: LCD only (1), status only (2), or both (0)

Type

int

Default value

0x0

display_poreset_button

Display power-on reset button

Type

bool

Default value

0x0

idler.delay_ms

GUI update period in ms

Type

int

Default value

0x32

idler.has_gui

GUI is enabled

Type

bool

Default value

0x1

is_heterogeneous_cluster

Is Heterogeneous cluster

Type

bool

Default value

0x0

lcd_height_param

LCD Height

Type

int

Default value

0x258

lcd_width_param

LCD Width

Type

int

Default value

0x320

mcp_name

MCP name

Type

string

Default value

MCP: Cortex-M7

num_cps

Number of Control Processors

Type

int

Default value

0x1

per_core_clock

Per-core clock connection

Type

bool

Default value

0x0

platform_name

Platform Name

Type

string

Default value

Application Processors

rate_limit-enable

Rate limit simulation.

Type

bool

Default value

0x0

recorder.checkInstructionCount

check instruction count in recording file against actual instruction count during playback

Type

bool

Default value

0x1

recorder.playbackDivider.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

recorder.playbackDivider.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

recorder.playbackFileName

playback filename (empty std::string disables playback)

Type

string

Default value**recorder.recordingDivider.div**

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

recorder.recordingDivider.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

recorder.recordingFileName

recording filename (empty std::string disables recording)

Type

string

Default value**recorder.recordingTimeBase**

timebase in 1/s (relative to the master clock (e.g. 100000000 means 10 nanoseconds resolution simulated time for a 1Hz master clock)) to be used for recording (higher values -> higher time resolution, playback time base is always taken from the playback file)

Type

int

Default value

0x5f5e100

recorder.verbose

enable verbose messages (1=normal, 2=even more)

Type

int

Default value

0x0

scp_name

SCP name

Type

string

Default value

SCP: Cortex-M7

shutdown_pixel_enable

Shutdown pixel enable. Use to trigger simulation shutdown when a specific pixel reaches the given target RGB value

Type

bool

Default value

0x0

shutdown_pixel_rgb

Shutdown pixel target RGB value 0xRRGGBB

Type

int

Default value

0xffffffff

shutdown_pixel_x

Shutdown pixel X co-ordinate (0 is left)

Type

int

Default value

0x0

shutdown_pixel_y

Shutdown pixel Y co-ordinate (0 is top)

Type

int

Default value

0x0

trap_key

Trap key that works with left Ctrl to toggle mouse display.

Type

int

Default value

0x4a

window_title

Window title(%cpu% will be replaced by css_spec)

Type

string

Default value

Fast Models - %cpu%

3.7.60 WarningMemory

Memory that prints warnings, and RAZ/WIs or aborts. This model is written in C++.

Ports for WarningMemory

Table 3-782: Ports

Name	Protocol	Type	Description
pvbust	PVBus	Slave	Bus slave interface

Parameters for WarningMemory

abort_on_reads

Generate Abort on reads

Type

bool

Default value

0x0

abort_on_writes

Generate Abort on writes

Type

bool

Default value

0x0

read_data

Data to return on reads, if not aborting

Type

int

Default value

0x0

warn_on_reads

Generate Warn on reads

Type

bool

Default value

0x1

warn_on_writes

Generate Warn on writes

Type

bool

Default value

0x1

warning

Warning string

Type

string

Default value
Invalid access

3.7.61 v8EmbeddedCrossTrigger_Matrix

v8 Embedded Cross Trigger Matrix. This model is written in C++.

About v8EmbeddedCrossTrigger_Matrix

This is a model of a platform-level *Cross Trigger Matrix* (CTM) for connection to the *Cross Trigger Interface* (CTI) ports provided on Arm®v8-A processors in Fast Models. The combination of the CTI and the CTM provides an architectural model of the Coresight™ embedded triggering system.

A single instance of the v8EmbeddedCrossTrigger_Matrix component supports up to four clusters, each containing four cores. For example:

```
cluster0.cti[0] => v8ect.cti[0];
cluster0.cti[1] => v8ect.cti[1];
cluster0.cti[2] => v8ect.cti[2];
cluster0.cti[3] => v8ect.cti[3];
...
cluster4.cti[3] => v8ect.cti[12];
cluster4.cti[3] => v8ect.cti[13];
cluster4.cti[3] => v8ect.cti[14];
cluster4.cti[3] => v8ect.cti[15];
```

Ports for v8EmbeddedCrossTrigger_Matrix

Table 3-783: Ports

Name	Protocol	Type	Description
cti[36]	v8EmbeddedCrossTrigger_controlprotocol	Slave	-

Parameters for v8EmbeddedCrossTrigger_Matrix

has_CTIAUTHSTATUS
Has the optional CTIAUTHSTATUS register
Type
bool
Default value
0x1

has_CTIDEVID_INOUT
Has the option of input gate in cross trigger matrix
Type
bool
Default value
0x1

number-of-channels

Number of channels in cross trigger matrix

Type

int

Default value

0x4

3.8 Scheduler components

This section describes the Scheduler components.

3.8.1 AsyncSignal

Allows to cleanly schedule events from non-simulation threads onto the simulation thread. This model is written in C++.

Ports for AsyncSignal

Table 3-784: Ports

Name	Protocol	Type	Description
async_callback	AsyncSignalCallback	Master	This port emits a call to signal() on the simulation thread asynchronously after async_control.signal() has been called.
async_control	AsyncSignalControl	Slave	Non-simulation threads call signal() on this port in order to schedule an event: a call to async_callback.signal() on the simulation thread.

3.8.2 SchedulerInterface

A SchedulerInterface instance allows access to the Fast Models scheduler. This model is written in LISA+.

Iris and MTI instances for SchedulerInterface

This model has the following Iris instances:

Table 3-785: SchedulerInterface Iris instances

InstanceName	ComponentName
SchedulerInterface	SchedulerInterface

SchedulerInterface contains the following CADI targets:

- SchedulerInterface

Ports for SchedulerInterface

Table 3-786: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Clock frequency for waitTicks() function.
control	SchedulerInterfaceControl	Slave	Scheduler interface. Allows to: - wait for time

3.8.3 SchedulerThread

A SchedulerThread instance represents a co-routine thread in the simulation. This model is written in LISA+.

Iris and MTI instances for SchedulerThread

This model has the following Iris instances:

Table 3-787: SchedulerThread Iris instances

InstanceName	ComponentName
SchedulerThread	SchedulerThread

SchedulerThread contains the following CADI targets:

- SchedulerThread

Ports for SchedulerThread

Table 3-788: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Clock frequency for waitTicks() function.
control	SchedulerThreadControl	Slave	SchedulerThread control. Masters use this to: - control the thread (wait etc) - implement the actual thread function threadProc()

3.8.4 SchedulerThreadEvent

A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on. This model is written in LISA+.

Iris and MTI instances for SchedulerThreadEvent

This model has the following Iris instances:

Table 3-789: SchedulerThreadEvent Iris instances

InstanceName	ComponentName
SchedulerThreadEvent	SchedulerThreadEvent

SchedulerThreadEvent contains the following CADI targets:

- SchedulerThreadEvent

Ports for SchedulerThreadEvent

Table 3-790: Ports

Name	Protocol	Type	Description
control	SchedulerThreadEventControl	Slave	SchedulerThreadEvent control. Masters use this to: - wait for this event - notify waiters that the event happened

3.9 Signals components

This section describes the Signals components.

3.9.1 AndGate

And Gate. This model is written in LISA+.

Iris and MTI instances for AndGate

This model has the following Iris instances:

Table 3-791: AndGate Iris instances

InstanceName	ComponentName
AndGate	AndGate

AndGate contains the following CADI targets:

- AndGate

About AndGate

This component implements a logical AND of two signal input ports to generate a single output signal. For example, you can use it to combine two interrupt signals.

Ports for AndGate

Table 3-792: Ports

Name	Protocol	Type	Description
input[2]	Signal	Slave	2 input signals to be AND'ed.
output	Signal	Master	AND'ed output signal.

3.9.2 FrequencyProbe_cpp

Clock Frequency observer. This model is written in C++.

Iris and MTI instances for FrequencyProbe_cpp

This model has the following Iris instances:

Table 3-793: FrequencyProbe_cpp Iris instances

InstanceName	ComponentName
FrequencyProbe_cpp	FrequencyProbe

FrequencyProbe_cpp contains the following CADI targets:

- FrequencyProbe

Ports for FrequencyProbe_cpp

Table 3-794: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	-
freq_changed	ValueState	Master	-

Parameters for FrequencyProbe_cpp

diagnostics

Diagnostics

Type

int

Default value

0x0

3.9.3 LabellerMasterIdExtendedIdUserFlag

Allows the modification of MasterID, ExtendedID and UserFlags attributes of PVBus transactions. This model is written in LISA+.

Iris and MTI instances for LabellerMasterIdExtendedIdUserFlag

This model has the following Iris instances:

Table 3-795: LabellerMasterIdExtendedIdUserFlag Iris instances

InstanceName	ComponentName
LabellerMasterIdExtendedIdUserFlag	LabellerMasterIdExtendedIdUserFlag
LabellerMasterIdExtendedIdUserFlag.pvbuslogger	PVBusLogger
LabellerMasterIdExtendedIdUserFlag.pvbuslogger.mapper	PVBusMapper
LabellerMasterIdExtendedIdUserFlag.pvbusmodifier	PVBusMapper

This model has the following MTI trace components:

Table 3-796: LabellerMasterIdExtendedIdUserFlag MTI instances

InstanceName	ComponentName
LabellerMasterIdExtendedIdUserFlag.pvbuslogger	PVBusLogger

InstanceName	ComponentName
LabellerMasterIdExtendedIdUserFlag.pvbuslogger.mapper	PVBusMapper
LabellerMasterIdExtendedIdUserFlag.pvbusmodifier	PVBusMapper

LabellerMasterIdExtendedIdUserFlag contains the following CADI targets:

- LabellerMasterIdExtendedIdUserFlag
- PVBusLogger

Ports for LabellerMasterIdExtendedIdUserFlag

Table 3-797: Ports

Name	Protocol	Type	Description
pvbus_m	PVBus	Master	Output with modified properties.
pvbus_s	PVBus	Slave	Unmodified input.

Parameters for LabellerMasterIdExtendedIdUserFlag

extendedid

ExtendedID value to be applied to transactions.

Type

int

Default value

0x0

extendedid_mask

Mask used to determine which bits of extendedid parameter to be set in the transactions ExtendedID attribute. 0xFFFFFFFFFFFFFFFF will overwrite all the incoming ExtendedID bits with the value of the extendedid parameter, 0x0 will overwrite none.

Type

int

Default value

0x0

masterid

MasterID value to be applied to transactions.

Type

int

Default value

0x0

masterid_mask

Mask used to determine which bits of masterid parameter to be set in the transactions MasterID attribute. 0xFFFFFFFF will overwrite all the incoming MasterID bits with the value of the masterid parameter, 0x0 will overwrite none.

Type

int

Default value

0x0

pvbuslogger.trace_debug

Enable tracing of debug transactions

Type

bool

Default value

0x0

pvbuslogger.trace_snoops

Enable tracing of ACE snoop requests

Type

bool

Default value

0x0

userflags

UserFlags value to be applied to transactions.

Type

int

Default value

0x0

userflags_mask

Mask used to determine which bits of userflags parameter to be set in the transactions UserFlags attribute. 0xFFFFFFFF will overwrite all the incoming UserFlags bits with the value of the userflags parameter, 0x0 will overwrite none.

Type

int

Default value

0x0

3.9.4 LabellerUserSignals

This model is written in LISA+.

Iris and MTI instances for LabellerUserSignals

This model has the following Iris instances:

Table 3-798: LabellerUserSignals Iris instances

InstanceName	ComponentName
LabellerUserSignals	LabellerUserSignals
LabellerUserSignals.pvbuslogger	PVBusLogger
LabellerUserSignals.pvbuslogger.mapper	PVBusMapper
LabellerUserSignals.pvbusmodifier	PVBusMapper

This model has the following MTI trace components:

Table 3-799: LabellerUserSignals MTI instances

InstanceName	ComponentName
LabellerUserSignals.pvbuslogger	PVBusLogger
LabellerUserSignals.pvbuslogger.mapper	PVBusMapper
LabellerUserSignals.pvbusmodifier	PVBusMapper

LabellerUserSignals contains the following CADI targets:

- LabellerUserSignals
- PVBusLogger

Ports for LabellerUserSignals

Table 3-800: Ports

Name	Protocol	Type	Description
pvbus_m	PVBus	Master	Output with modified UserFlags.
pvbus_s	PVBus	Slave	Unmodified input.

Parameters for LabellerUserSignals

pvbuslogger.trace_debug

Enable tracing of debug transactions

Type

bool

Default value

0x0

pvbuslogger.trace_snoops

Enable tracing of ACE snoop requests

Type

bool

Default value

0x0

user

User signal to be applied to transactions

Type

int

Default value

0x0

3.9.5 OrGate

Or Gate. This model is written in LISA+.

Iris and MTI instances for OrGate

This model has the following Iris instances:

Table 3-801: OrGate Iris instances

InstanceName	ComponentName
OrGate	OrGate

OrGate contains the following CADI targets:

- OrGate

About OrGate

This component implements a logical OR of two signal input ports to generate a single output signal. For example, you can use this component to combine two interrupt signals.

Ports for OrGate

Table 3-802: Ports

Name	Protocol	Type	Description
input[16]	Signal	Slave	16 input signals to be OR'ed.
output	Signal	Master	OR'ed output signal.

3.9.6 SGSignalBuffer

Buffer to synchronise SystemGenerator Signal setValue() calls. This model is written in LISA+.

Iris and MTI instances for SGSignalBuffer

This model has the following Iris instances:

Table 3-803: SGSignalBuffer Iris instances

InstanceName	ComponentName
SGSignalBuffer	SGSignalBuffer

SGSignalBuffer contains the following CADI targets:

- SGSignalBuffer



Variants of this component also exist with multiple input and output ports.

Ports for SGSignalBuffer

Table 3-804: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Clock rate to release buffered signals.
in	Signal	Slave	Signal in.
out	Signal	Master	Buffered signal out.

3.9.7 SignalDriver

Drives signal port based on parameter, register or bus slave port. This model is written in LISA+.

Iris and MTI instances for SignalDriver

This model has the following Iris instances:

Table 3-805: SignalDriver Iris instances

InstanceName	ComponentName
SignalDriver	SignalDriver
SignalDriver.pvbuslave	PVBusSlave

This model has the following MTI trace components:

Table 3-806: SignalDriver MTI instances

InstanceName	ComponentName
SignalDriver	SignalDriver
SignalDriver.pvbuslave	PVBusSlave

SignalDriver contains the following CADI targets:

- SignalDriver

Ports for SignalDriver

Table 3-807: Ports

Name	Protocol	Type	Description
pvbus_s	PVBus	Slave	-
signal_out	Signal	Master	-

Parameters for SignalDriver

param_input

Drive signal_out port with this parameter value

Type

bool

Default value

0x0

3.9.8 SignalInverter_cpp

This model is written in C++.

Iris and MTI instances for SignalInverter_cpp

This model has the following Iris instances:

Table 3-808: SignalInverter_cpp Iris instances

InstanceName	ComponentName
SignalInverter_cpp	SignalInverter

SignalInverter_cpp contains the following CADI targets:

- SignalInverter

Ports for SignalInverter_cpp

Table 3-809: Ports

Name	Protocol	Type	Description
sig_in	Signal	Slave	-
sig_out	Signal	Master	-

Name	Protocol	Type	Description
sig_out_invert	Signal	Master	-

3.9.9 SignalLogger

Traces signal activity. This model is written in LISA+.

Iris and MTI instances for SignalLogger

This model has the following Iris instances:

Table 3-810: SignalLogger Iris instances

InstanceName	ComponentName
SignalLogger	SignalLogger

This model has the following MTI trace components:

Table 3-811: SignalLogger MTI instances

InstanceName	ComponentName
SignalLogger	SignalLogger

SignalLogger contains the following CADI targets:

- SignalLogger

Ports for SignalLogger

Table 3-812: Ports

Name	Protocol	Type	Description
in	Signal	Slave	Input signal port.
out	Signal	Master	Output signal port.

Parameters for SignalLogger

forward_signal

If true, trace signal and forward signal from 'in' to 'out'. If false, trace signal only without driving 'out' port

Type

bool

Default value

0x1

3.9.10 Value64Logger

Traces value activity. This model is written in LISA+.

Iris and MTI instances for Value64Logger

This model has the following Iris instances:

Table 3-813: Value64Logger Iris instances

InstanceName	ComponentName
Value64Logger	Value64Logger

This model has the following MTI trace components:

Table 3-814: Value64Logger MTI instances

InstanceName	ComponentName
Value64Logger	Value64Logger

Value64Logger contains the following CADI targets:

- Value64Logger

Ports for Value64Logger

Table 3-815: Ports

Name	Protocol	Type	Description
in	Value_64	Slave	Input signal port.
out	Value_64	Master	Output signal port.

3.9.11 ValueLogger

Traces value activity. This model is written in LISA+.

Iris and MTI instances for ValueLogger

This model has the following Iris instances:

Table 3-816: ValueLogger Iris instances

InstanceName	ComponentName
ValueLogger	ValueLogger

This model has the following MTI trace components:

Table 3-817: ValueLogger MTI instances

InstanceName	ComponentName
ValueLogger	ValueLogger

ValueLogger contains the following CADI targets:

- ValueLogger

Ports for ValueLogger

Table 3-818: Ports

Name	Protocol	Type	Description
in	Value	Slave	Input signal port.
out	Value	Master	Output signal port.

3.9.12 WideAndGate_cpp

And Gate with up to 8 inputs. This model is written in C++.

Iris and MTI instances for WideAndGate_cpp

This model has the following Iris instances:

Table 3-819: WideAndGate_cpp Iris instances

InstanceName	ComponentName
WideAndGate_cpp	WideAndGate

WideAndGate_cpp contains the following CADI targets:

- WideAndGate

Ports for WideAndGate_cpp

Table 3-820: Ports

Name	Protocol	Type	Description
input[8]	Signal	Slave	-
output	Signal	Master	-

Parameters for WideAndGate_cpp

diagnostics

Diagnostics

Type

int

Default value

0x0

3.9.13 WideOrGate_cpp

Or Gate with up to 8 inputs. This model is written in C++.

Iris and MTI instances for WideOrGate_cpp

This model has the following Iris instances:

Table 3-821: WideOrGate_cpp Iris instances

InstanceName	ComponentName
WideOrGate_cpp	WideOrGate

WideOrGate_cpp contains the following CADI targets:

- WideOrGate

Ports for WideOrGate_cpp

Table 3-822: Ports

Name	Protocol	Type	Description
input[8]	Signal	Slave	-
output	Signal	Master	-

Parameters for WideOrGate_cpp

diagnostics

Diagnostics

Type

int

Default value

0x0

3.10 SystemIP components

This section describes the SystemIP components.

The major SystemIP components are:

- Input/output devices.
- Memory, including flash.
- Ethernet controller.
- Interrupt controllers.
- Static and dynamic memory controllers.
- Audio interface.
- Programmable clock generators.

These components are software implementations of specific hardware functionality.

3.10.1 AHCI_SATA

AHCI controller with attached SATA disks and PCIe interface. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-823: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for AHCI_SATA

This model has the following Iris instances:

Table 3-824: AHCI_SATA Iris instances

InstanceName	ComponentName
AHCI_SATA	AHCI_SATA
AHCI_SATA.ahci_master	PVBusMaster
AHCI_SATA.register_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-825: AHCI_SATA MTI instances

InstanceName	ComponentName
AHCI_SATA	ExportTestAHCI_SATA
AHCI_SATA.ahci_master	PVBusMaster
AHCI_SATA.register_slave	PVBusSlave

AHCI_SATA contains the following CADI targets:

- AHCI_SATA

Ports for AHCI_SATA

Table 3-826: Ports

Name	Protocol	Type	Description
ahci_dma_m	PVBus	Master	-
client_s	PCIDevice2ClientProtocol	Slave	-
pvbus	PVBus	Slave	-

Parameters for AHCI_SATA

force_mode
Force disk to report support for at most PIO/DMA/NCQ mode (only for testing/bring-up purposes). PIO mode is always supported. Use NCQ for maximum performance (default).
Type
string
Default value
NCQ
image_path
Comma separated list of zero or more disk images (up to 32). Each image represents one SATA disk which is connected to one port of the AHCI controller. Empty list elements are allowed and result in a SATA port which has no disk attached. Empty string (default) means: One SATA port with no disk attached. Use 'truncate -s 4T disk.img' to create a 4 TByte sparse image. Use 'dd if=/dev/zero of=disk.img bs=1M count=42' to create a 42 MByte non-sparse image.
Type
string
Default value
run_async
Do host I/O in a background thread asynchronously. Enabling this makes the simulation non-deterministic and may or may not improve performance. Default is 'false' (do all disk accesses synchronously).
Type
bool
Default value
0x0

3.10.2 AddressTranslationUnit

Address Translator Unit. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-827: IP revisions support

Revision	Quality level
0.5	Alpha support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for AddressTranslationUnit

This model has the following Iris instances:

Table 3-828: AddressTranslationUnit Iris instances

InstanceName	ComponentName
AddressTranslationUnit	AddressTranslationUnit
AddressTranslationUnit.ATU_BusMapper	PVBusMapper
AddressTranslationUnit.apb	PVBusSlave

This model has the following MTI trace components:

Table 3-829: AddressTranslationUnit MTI instances

InstanceName	ComponentName
AddressTranslationUnit.ATU_BusMapper	PVBusMapper
AddressTranslationUnit.apb	PVBusSlave

AddressTranslationUnit contains the following CADI targets:

- AddressTranslationUnit

Ports for AddressTranslationUnit

Table 3-830: Ports

Name	Protocol	Type	Description
apb	PVBus	Slave	-
irq_out	Signal	Master	-
pvbus_m	PVBus	Master	-
pvbus_s	PVBus	Slave	-
reset_in	Signal	Slave	-

Parameters for AddressTranslationUnit

ATUNTR

Number of translation regions (1=2, 2=4, 3=8, 4=16, and 5=32)

Type

int

Default value

0x5

ATUPAW

Physical address width (0=32, 1=36, 2=40, 3=44, 4=48, 5=52, 6=56, 7=64 bits). No impact on the PVBUS transactions.

Type

int

Default value

0x5

ATUPS

Selects the page size granularity in bytes (0xC=4096, 0xD=8192, and 0xE=16384)
(Default=0xD)

Type

int

Default value

0xd

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type

int

Default value

0x0

3.10.3 BP141_TZMA

PrimeCell Infrastructure AMBA 3 AXI TrustZone Memory Adapter. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-831: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for BP141_TZMA

This model has the following Iris instances:

Table 3-832: BP141_TZMA Iris instances

InstanceName	ComponentName
BP141_TZMA	BP141_TZMA
BP141_TZMA.pvbusrange_0	PVBusRange
BP141_TZMA.pvbusrange_0.pvbus_mapper	PVBusMapper
BP141_TZMA.tzswitch_0	TZSwitch
BP141_TZMA.tzswitch_0.pvbus_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-833: BP141_TZMA MTI instances

InstanceName	ComponentName
BP141_TZMA.pvbusrange_0.pvbus_mapper	PVBusMapper
BP141_TZMA.tzswitch_0.pvbus_mapper	PVBusMapper

BP141_TZMA contains the following CADI targets:

- BP141_TZMA
- PVBusRange
- TZSwitch

About BP141_TZMA

BP141_TZMA permits a single physical memory cell of up to 2 MB to be shared between a secure and non-secure storage area. The partitioning between these areas is flexible.

This component routes transactions according to the following:

- The memory region that they are attempting to access.
- The security mode of the transaction.

The BP141_TZMA fixes the base address of the secure region to the base address of the decode space. It uses the R0SIZE [9:0] input to configure the size of the secure region in 4KB increments up to a maximum of 2 MB.

TZMEMSIZE is the maximum addressing range of the memory as defined by that parameter. By default, TZMEMSIZE is set to 2MB. In the following table, AxADDR is the offset address that the transactions want to access.

Table 3-834: BP141_TZMA security control

AxADDR	Memory Region	Non-secure Transfer	Secure Transfer
AxADDR < R0Size	Secure, R0	Illegal	Legal
R0SIZE <= AxADDR and AxADDR < TZMEMSIZE	Non-secure, R1	Legal	Legal
AxADDR => TZMEMSIZE	No access	Illegal	Illegal

Ports for BP141_TZMA

Table 3-835: Ports

Name	Protocol	Type	Description
pv_output	PVBus	Master	Routed PVBus output
pvbus	PVBus	Slave	Bus slave interface.
R0Size	Value	Slave	A software interface that is driven from the TrustZone Protection Controller (TZPC), setting the secure region size by bits[9:0].

Parameters for BP141_TZMA

TZMEMSIZE

Addressable range of device

Type
int

Default value
0x200000

TZSECROMSIZE
Default secure size

Type
int

Default value
0x200

TZSEGSIZE
Segment size

Type
int

Default value
0x1000

pvbusrange_0.range
Addressable range routed to pvbus_port_a

Type
int

Default value
0x0

3.10.4 BP147_TZPC

TrustZone Protection Controller. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-836: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for BP147_TZPC

This model has the following Iris instances:

Table 3-837: BP147_TZPC Iris instances

InstanceName	ComponentName
BP147_TZPC	BP147_TZPC
BP147_TZPC.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-838: BP147_TZPC MTI instances

InstanceName	ComponentName
BP147_TZPC.busslave	PVBusSlave

BP147_TZPC contains the following CADI targets:

- BP147_TZPC

About BP147_TZPC

BP147_TZPC provides a software interface to the protection bits in a secure system in a TrustZone® design.

Ports for BP147_TZPC

Table 3-839: Ports

Name	Protocol	Type	Description
bus_in_s	PVBus	Slave	Slave port for register access.
TZPCDECPROT0	Value	Master	Output decode protection 0 status.
TZPCDECPROT1	Value	Master	Output decode protection 1 status.
TZPCDECPROT2	Value	Master	Output decode protection 2 status.
TZPCR0SIZE	Value	Master	Output secure RAM region size.

3.10.5 CCI400

Cache Coherent Interconnect for AXI4 ACE. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-840: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CCI400

This model has the following Iris instances:

Table 3-841: CCI400 Iris instances

InstanceName	ComponentName
CCI400	CCI400
CCI400.cciinterconnect	PVCache
CCI400.cciregisters	CCIRegisters
CCI400.cciregisters.clocktimer	ClockTimerThread
CCI400.cciregisters.clocktimer.timer	ClockTimerThread64
CCI400.cciregisters.clocktimer.timer.thread	SchedulerThread
CCI400.cciregisters.clocktimer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-842: CCI400 MTI instances

InstanceName	ComponentName
CCI400	CCI400

CCI400 contains the following CADI targets:

- CCI400

About CCI400

- If you disable the `cache_state_modelled` parameter, this component has negligible performance impact. If you enable `cache_state_modelled`, it adds significant cost to throughput for coherent transactions.
- This model implements the slave interface Shareable Override Register, which can be read and written, but it has no functionality.

ACE limitation

AXI Coherency Extensions (ACE) are extensions to AXI4 that support system-level cache coherency between multiple clusters. The ACE cache models in the Arm® Cortex®-A15 and Cortex-A7, and the ACE support in the CCI-400 have the limitation that they only process one transaction at a time. Normally, the simulation processes each transaction to completion before allowing any master to generate another transaction. However, in the following situation the simulation might fail. If a SystemC bus slave calls `wait()` while it is processing a transaction, this call might allow another master to issue another transaction that passes through the CCI-400 or the Cortex-A15 or Cortex-A7 caches. This situation could happen if a SystemC bus master running in another thread is connected to one of the ACE-lite ports on the CCI-400.

Ports for CCI400

Table 3-843: Ports

Name	Protocol	Type	Description
acchannelen	Value	Slave	For each upstream port, determine if it is enabled or not with respect to snoop requests.
barrierterminate	Value	Slave	For each downstream port, determine if barriers are terminated at that port.

Name	Protocol	Type	Description
broadcastcachemain	Value	Slave	For each downstream port, determine if broadcast cache maintenance operations are forwarded down that port. A three bit signal but as the model only have a single downstream port, setting any of the bits will make it work.
bufferableoverride	Value	Slave	For each downstream port, determine if all transactions are forced to non-bufferable (AWCACHE[0] is forced to 0).
clk_in	ClockSignal	Slave	Clock signal for cciregisters
errorirq	Signal	Master	A signal stating that the imprecise error register is nonzero.
evntcntoverflow[5]	Signal	Master	When an event counter overflows, it sets the corresponding signal.
lint_ace_3_reset_state	Signal	Slave	This port can be connected to the reset signals of the system attached to the pvbus_s_ace_3 port.
lint_ace_4_reset_state	Signal	Slave	This port can be connected to the reset signals of the system attached to the pvbus_s_ace_4 port.
periphbase	Value_64	Slave	This port sets the base address of the private peripheral region.
pvbus_m	PVBus	Master	Master port for all downstream memory accesses.
pvbus_s_ace_3	PVBus	Slave	ACE-capable slave ports.
pvbus_s_ace_4	PVBus	Slave	ACE-capable slave ports.
pvbus_s_ace_lite_plus_dvm_0	PVBus	Slave	Memory bus interface that implements ACE lite and DVM protocol.
pvbus_s_ace_lite_plus_dvm_1	PVBus	Slave	Memory bus interface that implements ACE lite and DVM protocol.
pvbus_s_ace_lite_plus_dvm_2	PVBus	Slave	Memory bus interface that implements ACE lite and DVM protocol.
reset_in	Signal	Slave	Signal to reset the CCI.
reset_state_of_ace_lite_ports[3]	Signal	Slave	This port can be connected to the reset signals of the system attached to ACE-Lite ports 0,1,2

Parameters for CCI400

acchannelen

For each upstream port, determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x1f

barrierterminate

For each downstream port, determine if barriers will be terminated at that port.

Type

int

Default value

0x7

broadcastcachemain

For each downstream port a bit determines if broadcast cache maintenance operations are forwarded down that port.

Type

int

Default value

0x0

bufferableoverride

For each downstream port, determine if all transactions will be forced to non-bufferable.

Type

int

Default value

0x0

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

force_on_from_start

Enables snooping on upstream ports from the start of simulation. The CCI will normally start up with snooping disabled, however, using this parameter we allow the model to start with snooping enabled without SW drivers programming the CCI. This is only setup at simulation reset and not at signal reset. If the upstreams can ever be held in reset then you *must* connect: `reset_state_of_ace_lite_ports[]`, `lint_ace_3_reset_state`, `lint_ace_4_reset_state`, so that it knows when to disable snoops to the upstream systems. Otherwise, the upstream system will receive snoop messages whilst in reset and complain that it 'received a snoop request whilst it was in reset'.

Type

bool

Default value

0x0

is_downstream_domain_boundary_for_far_atomic

This interconnect is at the last stage of the domain boundary.

Type

bool

Default value

0x0

log_enabled
Enable log messages from the CCI register file. Log level 0 means do not print anything, 1 means print only access violations, 2 means also print writes, 3 means print reads as well.
Type
int
Default value
0x1

periphbase
Value for PERIPHBASE. Only bits [39:16] are used. This value may be overridden by an input on the periphbase port
Type
int
Default value
0x2c000000

revision
Revision of the CCI400
Type
string
Default value
r0p0

3.10.6 CCI500

Cache Coherent Interconnect. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-844: IP revisions support

Revision	Quality level
r0p0	Full support
r0p2	Full support
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CCI500

This model has the following Iris instances:

Table 3-845: CCI500 Iris instances

InstanceName	ComponentName
CCI500	CCI500
CCI500.pvbus_register_file_s[0]	PVBusSlave
CCI500.upstream[0]	PVBusSlave
CCI500.upstream[1]	PVBusSlave
CCI500.upstream[2]	PVBusSlave
CCI500.upstream[3]	PVBusSlave
CCI500.upstream[4]	PVBusSlave
CCI500.upstream[5]	PVBusSlave
CCI500.upstream[6]	PVBusSlave

This model has the following MTI trace components:

Table 3-846: CCI500 MTI instances

InstanceName	ComponentName
CCI500	CCI500
CCI500.pvbus_register_file_s[0]	PVBusSlave
CCI500.upstream[0]	PVBusSlave
CCI500.upstream[1]	PVBusSlave
CCI500.upstream[2]	PVBusSlave
CCI500.upstream[3]	PVBusSlave
CCI500.upstream[4]	PVBusSlave
CCI500.upstream[5]	PVBusSlave
CCI500.upstream[6]	PVBusSlave

CCI500 contains the following CADI targets:

- CCI500

About CCI500

The LISA file declares seven upstream ports. You can configure these ports with `num_ace_ports` and `num_ace_lite_ports`. The bottom `num_ace_lite_ports` are ACE-Lite+DVM. The next `num_ace_ports` are ACE. Any remaining ports are ignored. If transactions are made on them, then warnings are produced. For example, if `num_ace_ports = 1` and `num_ace_lite_ports = 1` then `pvbus_s[1]` is ACE, `pvbus_s[0]` is ACE-Lite+DVM and `pvbus_s[6-2]` are considered not to exist.

Differences between the model and the RTL

Address Decoder

- Only supports striping down to 4KiB.
- If the address decoder aborts the access, returns `SLVERR` rather than `DECERR`.

Interfaces

The model does not implement the Q-Channel and P-Channel interfaces.

Performance Monitoring Unit

- PMU counters recognize only a few event sources:
 - Slave interface events:

3	ReadOnce.
4	ReadClean, ReadShared, ReadNotSharedDirty, ReadUnique.
5	MakeUnique, CleanUnique.
6	CleanInvalid, CleanShared, MakeInvalid.
7	DVM transaction received from upstream.
9	Read data that is satisfied by a snoop request.
 - No events are implemented for the global events or for the master events.
- The PMU does not implement the event bus (EVENTBUS).

Register access

The register file only supports 32-bit accesses to its registers. Later versions of the CCI500 hardware support full write strobes to the register file. This limitation means that byte and halfword accesses work on the hardware but not on this version of the component.

Snoop filter RAMs

Snoop filter RAMs are not modeled. The Status Register fields that relate to the power state of the Snoop filter RAMs are undefined.

Ports for CCI500

Table 3-847: Ports

Name	Protocol	Type	Description
acchannelensx[7]	Value	Slave	ACCHANNELENSx represents the ports ACCHANNELENS0..ACCHANNELENS7 on the RTL (assuming there are seven upstream ports). * each upstream ACE port 'y' (pvbus_s[y]) has a two bit ACCHANNELENSx * bit 0 == 0 -- DVM messages are disabled from being sent to this interface * bit 1 == 0 -- Snoop messages are disabled from being sent to this interface * each upstream ACE-Lite port 'z' (pvbus_s[z]) has a one bit ACCHANNELENSx * bit 0 == 0 -- DVM messages are disabled from being sent to this interface In the model, as we support a variety of configurations with a single LISA file then each port will behave as though it is one bit or two bit as appropriate. If you send a value that cannot be represented, given the width of the port, then the CCI model will halt and produce a fatal error. The assumed values of these are set by parameters until they are driven, so you need not drive them if they are constant. In the RTL, these signals are sampled at reset. Due to ordering issues w.r.t. reset() on different components then we cannot do that. Instead the signals are sampled at first transaction. Thus any controller that is producing these signals has to hold them constant for long enough. AC channel enables.

Name	Protocol	Type	Description
address_decoder	CCI500_ AddressDecoderProtocol	Master	An address decoder can be attached to the address_decoder port to choose which pvbuss port a downstream transaction will go out of. If you do not connect an address decoder then all transactions will go out of port 0.
dbgen	Signal	Slave	Invasive debug enable.
errirq	Signal	Master	Indicates that an error response, DECERR or SLVERR, is received on the RRESP, BRESP, or CRRESP input signals, and it cannot be signaled precisely.
evntcntoverflow[8]	Signal	Master	Overflow flags for the PMU clock and counters.
niden	Signal	Slave	Non-invasive debug enable.
pvbus_m[6]	PVBus	Master	Bus master ports.
pvbus_ register_file_s	PVBus	Slave	The slave port of the register file.
pvbus_s[7]	PVBus	Slave	Bus slave ports.
reset_in	Signal	Slave	Reset the interconnect.
reset_state_ of_upstream_port[7]	Signal	Slave	Tell the interconnect the reset state of the upstream ports, this can be used by the interconnect to check some aspects of the reset sequencing. If you are using force_on_from_start then you _must_ connect these pins.
spiden	Signal	Slave	Secure invasive debug enable.
spniden	Signal	Slave	Secure privileged non-invasive debug enable.

Parameters for CCI500

acchannelens0

For upstream port 0 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens1

For upstream port 1 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens2

For upstream port 2 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens3

For upstream port 3 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens4

For upstream port 4 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens5

For upstream port 5 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens6

For upstream port 6 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

addr_width

The bit-width of the address that the CCI can accept.

Type

int

Default value

0x28

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

dbgen

Invasive debug enable. If true, enables the counting of PMU events.

Type

bool

Default value

0x1

enable_logger

Enable PVBUSLoggers for the downstream ports in the CCI model.

Type

bool

Default value

0x0

force_on_from_start

The interconnect will normally start up with snooping/DVM disabled. This parameter enables the upstream system of a port to be snooped if the upstream is not in reset and if ACCHANNELENSx allows it. No software driver for the interconnect is needed. Any port that could go into reset must have 'reset_state_of_upstream_port[]' reflect the reset state of that upstream system. Otherwise, the upstream system may receive snoop/DVM messages whilst in reset and may complain that it 'received a snoop request whilst it was in reset'. Do not use if software is directly controlling the interconnect. This option does not disavow the responsibility of the upstream system to clean any shared dirty data from its caches before going into reset.

Type

bool

Default value

0x0

niden

Whether non-secure events are allowed to be counted in the performance monitor

Type

bool

Default value

0x1

num_ace_lite_ports

The bottom num_ace_lite_ports are ACE-Lite+DVM.

Type

int

Default value

0x5

num_ace_ports

The top num_ace_ports are ACE and support full coherency.

Type

int

Default value

0x2

number_of_phantom_entries

Number of phantom entries in the cache. Phantom entries are used by certain cache operations to hold temporary data. Usually this should be left at the default value which is safe for all systems containing up to 32 masters.

Type

int

Default value

0x20

qos_threshold_upper

Reset value for the QoS threshold register.

Type

int

Default value

0xc

reentrancy_support

Must be one of: on/off/cacheglobal/env: 'on': hazard checking per cache line (normal mode), 'off': no hazard checking (use only for single master systems), 'cacheglobal': hazard checking globally for cache (not per cache line, testing feature, provokes more hazards than necessary), 'env' (or empty string): take value from FM_REENTRANCY_SUPPORT env var, if this is not set use 'on', default is 'env'

Type

string

Default value

env

spiden

Secure invasive debug enable. If both SPIDEN and DBGEN are high, enables the counting of both Non-secure and Secure events.

Type
bool

Default value
0x1

spniden

Whether secure and non-secure events are allowed to be counted in the performance monitor

Type
bool

Default value
0x1

version

The version of the interconnect. Allowed versions are:- r0p0, r0p2, r1p0

Type
string

Default value
r0p0

3.10.7 CCI550

Cache Coherent Interconnect for AXI4 ACE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-848: IP revisions support

Revision	Quality level
r0p0	Full support
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CCI550

This model has the following Iris instances:

Table 3-849: CCI550 Iris instances

InstanceName	ComponentName
CCI550	CCI550

InstanceName	ComponentName
CCI550.pvbus_register_file_s[0]	PVBusSlave
CCI550.upstream[0]	PVBusSlave
CCI550.upstream[1]	PVBusSlave
CCI550.upstream[2]	PVBusSlave
CCI550.upstream[3]	PVBusSlave
CCI550.upstream[4]	PVBusSlave
CCI550.upstream[5]	PVBusSlave
CCI550.upstream[6]	PVBusSlave

This model has the following MTI trace components:

Table 3-850: CCI550 MTI instances

InstanceName	ComponentName
CCI550	CCI550
CCI550.pvbus_register_file_s[0]	PVBusSlave
CCI550.upstream[0]	PVBusSlave
CCI550.upstream[1]	PVBusSlave
CCI550.upstream[2]	PVBusSlave
CCI550.upstream[3]	PVBusSlave
CCI550.upstream[4]	PVBusSlave
CCI550.upstream[5]	PVBusSlave
CCI550.upstream[6]	PVBusSlave

CCI550 contains the following CADI targets:

- CCI550

Differences between the model and the RTL

Address Decoder

- Only supports striping down to 4KiB.
- If the address decoder aborts the access, returns `SILVERR` rather than `DECERR`.

Interfaces

The model does not implement the Q-Channel and P-Channel interfaces.

Performance Monitoring Unit

- PMU counters recognize only a few event sources:
 - Slave interface events:

3	ReadOnce.
4	ReadClean, ReadShared, ReadNotSharedDirty, ReadUnique.
5	MakeUnique, CleanUnique.

- 6 CleanInvalid, CleanShared, MakeInvalid.
- 7 DVM transaction received from upstream.
- 9 Read data that is satisfied by a snoop request.

- No events are implemented for the global events or for the master events.
- The PMU does not implement the event bus (EVNTBUS).

Reset signal sampling

The configuration ports `acchannelensx[]` are sampled in the hardware when coming out of reset. In the model, these ports are sampled at the first transaction to a `pvbus_s` port or to the register file.

Status Register, change-pending, and DVM messages

The Status Register provides information on when the last transaction that could have observed an old value of a snoop or DVM enable has finished in the upstream system. Therefore a port that has been disabled can now have the system upstream of that port turned off. The model does not track DVM messages in the upstream system.

Snoop filter RAMs

- The CCI-550 hardware has a snoop filter that reduces the number of snoop requests that the interconnect has to make. The model does not have a snoop filter and could make more snoop requests than the hardware would. This difference has no programmer-visible effect.
- The Status Register fields that relate to the power state of the Snoop filter RAMs are undefined.

Registers

The following registers provide storage but have no effect on the model.

- QoS registers.
- Interface monitor registers. These registers are intended for silicon debug.

Ports for CCI550

Table 3-851: Ports

Name	Protocol	Type	Description
<code>acchannelensx[7]</code>	Value	Slave	The <code>acchannelensx[N]</code> pins are used to tell the interconnect if the upstream system will accept snoops and/or DVM messages.
<code>address_decoder</code>	CCI500_ AddressDecoderProtocol	Master	An address decoder can be attached to the <code>address_decoder</code> port to choose which <code>pvbus_s</code> port a downstream transaction will go out of. If you do not connect an address decoder then all transactions will go out of port 0.
<code>dbgen</code>	Signal	Slave	Invasive debug enable.
<code>errirq</code>	Signal	Master	Some async error was detected.
<code>evntcntoverflow[8]</code>	Signal	Master	The output interrupts of the event counters.
<code>niden</code>	Signal	Slave	Non-invasive debug enable.
<code>pvbus_m[7]</code>	PVBus	Master	The downstream master ports.
<code>pvbus_ register_file_s</code>	PVBus	Slave	The slave port of the register file.

Name	Protocol	Type	Description
pvbus_s[7]	PVBus	Slave	Bus slave ports.
reset_in	Signal	Slave	Reset the interconnect.
reset_state_of_upstream_port[7]	Signal	Slave	Tell the interconnect the reset state of the upstream ports, this can be used by the interconnect to check some aspects of the reset sequencing. If you are using force_on_from_start then you _must_ connect these pins.
sci_s[7]	SystemCoherencyInterface	Slave	The System Coherency Interface bus. For those upstream ports that have a corresponding bit set in the bitmap of si_system_coherency_interface then the corresponding sci_m port can be used to move the upstream system into and out of the coherency domain.
spiden	Signal	Slave	Secure privileged invasive debug enable.
spniden	Signal	Slave	Secure privileged non-invasive debug enable.

Parameters for CCI550

acchannelens0

For upstream port 0 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens1

For upstream port 1 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens2

For upstream port 2 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens3

For upstream port 3 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens4

For upstream port 4 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens5

For upstream port 5 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

acchannelens6

For upstream port 6 determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0x0

addr_width

The bit-width of the address that the CCI can accept.

Type

int

Default value

0x28

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

dbgen

Invasive debug enable. If true, enables the counting of PMU events.

Type

bool

Default value

0x1

enable_logger

Enable PVBUSLoggers for the downstream ports in the CCI model.

Type

bool

Default value

0x0

force_on_from_start

The interconnect will normally start up with snooping/DVM disabled. The parameter `si_system_coherency_interface` determines which connections are managed by the System Coherency Interface (SCI). For connections that are managed by SCI, then this parameter has no effect. For all other connections, this parameter enables the upstream system of a port to be snooped if the upstream is not in reset and if `ACCHANNELENSx` allows it. No software driver for the interconnect is needed. Any non-SCI port that could go into reset must have '`reset_state_of_upstream_port[]`' reflect the reset state of that upstream system. Otherwise, the upstream system may receive snoop/DVM messages whilst in reset and may complain that it 'received a snoop request whilst it was in reset'. Do not use if software is directly controlling the interconnect. This option does not disavow the responsibility of the upstream system to clean any shared dirty data from its caches before going into reset.

Type

bool

Default value

0x0

niden

Whether non-secure events are allowed to be counted in the performance monitor

Type

bool

Default value

0x1

num_ace_lite_ports

The bottom `num_ace_lite_ports` are ACE-Lite+DVM.

Type

int

Default value

0x5

num_ace_ports

The top `num_ace_ports` are ACE and support full coherency.

Type

int

Default value

0x2

number_of_phantom_entries

Number of phantom entries in the cache. Phantom entries are used by certain cache operations to hold temporary data. Usually this should be left at the default value which is safe for all systems containing up to 32 masters.

Type

int

Default value

0x20

qos_threshold_upper

Reset value for the QoS threshold register.

Type

int

Default value

0xc

reentrancy_support

Must be one of: on/off/cacheglobal/env: 'on': hazard checking per cache line (normal mode), 'off': no hazard checking (use only for single master systems), 'cacheglobal': hazard checking globally for cache (not per cache line, testing feature, provokes more hazards than necessary), 'env' (or empty string): take value from FM_REENTRANCY_SUPPORT env var, if this is not set use 'on', default is 'env'

Type

string

Default value

env

si0_qos_bw_regulator

For upstream port 0 determine if it has a BW regulator. The effect of QoS is not modelled and this parameter only alters some registers.

Type

bool

Default value

0x0

si1_qos_bw_regulator

For upstream port 1 determine if it has a BW regulator. The effect of QoS is not modelled and this parameter only alters some registers.

Type

bool

Default value

0x0

si2_qos_bw_regulator

For upstream port 2 determine if it has a BW regulator. The effect of QoS is not modelled and this parameter only alters some registers.

Type

bool

Default value

0x0

si3_qos_bw_regulator

For upstream port 3 determine if it has a BW regulator. The effect of QoS is not modelled and this parameter only alters some registers.

Type

bool

Default value

0x0

si4_qos_bw_regulator

For upstream port 4 determine if it has a BW regulator. The effect of QoS is not modelled and this parameter only alters some registers.

Type

bool

Default value

0x0

si5_qos_bw_regulator

For upstream port 5 determine if it has a BW regulator. The effect of QoS is not modelled and this parameter only alters some registers.

Type

bool

Default value

0x0

si6_qos_bw_regulator

For upstream port 6 determine if it has a BW regulator. The effect of QoS is not modelled and this parameter only alters some registers.

Type

bool

Default value

0x0

si_system_coherency_interface

This parameter tells the interconnect which upstream ports should be controlled by the System Coherency Interface. Each bit corresponds to an upstream port, bit 0 to upstream port 0, etc. If the SCI port is connected but si_system_coherency_interface disable its use then messages from the upstream will be ignored and software must manage the upstream system's entrance and exit of the coherency domain.

Type

int

Default value

0x0

spiden

Secure invasive debug enable. If both SPIDEN and DBGEN are high, enables the counting of both Non-secure and Secure events.

Type

bool

Default value

0x1

spniden

Whether secure and non-secure events are allowed to be counted in the performance monitor

Type

bool

Default value

0x1

version

The version of the interconnect. Allowed versions are:- r0p0, r1p0

Type

string

Default value

r0p0

3.10.8 CCN502

CCN502 Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-852: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CCN502

This model has the following Iris instances:

Table 3-853: CCN502 Iris instances

InstanceName	ComponentName
CCN502	CCN5XX
CCN502.bus_slave_ocm	PVBusSlave
CCN502.ccn502_hni_exclusive_monitor_0	PVBusExclusiveMonitor
CCN502.ccn502_hni_exclusive_monitor_0.bus_mapper	PVBusMapper
CCN502.ccn_cache	CCNCache
CCN502.ccn_cache.upstream[0]	PVBusSlave
CCN502.ccn_cache.upstream[10]	PVBusSlave
CCN502.ccn_cache.upstream[11]	PVBusSlave
CCN502.ccn_cache.upstream[12]	PVBusSlave
CCN502.ccn_cache.upstream[13]	PVBusSlave
CCN502.ccn_cache.upstream[14]	PVBusSlave
CCN502.ccn_cache.upstream[15]	PVBusSlave
CCN502.ccn_cache.upstream[16]	PVBusSlave
CCN502.ccn_cache.upstream[17]	PVBusSlave
CCN502.ccn_cache.upstream[18]	PVBusSlave
CCN502.ccn_cache.upstream[19]	PVBusSlave
CCN502.ccn_cache.upstream[1]	PVBusSlave
CCN502.ccn_cache.upstream[20]	PVBusSlave
CCN502.ccn_cache.upstream[21]	PVBusSlave
CCN502.ccn_cache.upstream[22]	PVBusSlave
CCN502.ccn_cache.upstream[23]	PVBusSlave
CCN502.ccn_cache.upstream[24]	PVBusSlave
CCN502.ccn_cache.upstream[25]	PVBusSlave
CCN502.ccn_cache.upstream[26]	PVBusSlave
CCN502.ccn_cache.upstream[27]	PVBusSlave
CCN502.ccn_cache.upstream[28]	PVBusSlave
CCN502.ccn_cache.upstream[29]	PVBusSlave
CCN502.ccn_cache.upstream[2]	PVBusSlave
CCN502.ccn_cache.upstream[30]	PVBusSlave
CCN502.ccn_cache.upstream[31]	PVBusSlave

InstanceName	ComponentName
CCN502.ccn_cache.upstream[32]	PVBusSlave
CCN502.ccn_cache.upstream[33]	PVBusSlave
CCN502.ccn_cache.upstream[34]	PVBusSlave
CCN502.ccn_cache.upstream[35]	PVBusSlave
CCN502.ccn_cache.upstream[36]	PVBusSlave
CCN502.ccn_cache.upstream[37]	PVBusSlave
CCN502.ccn_cache.upstream[38]	PVBusSlave
CCN502.ccn_cache.upstream[39]	PVBusSlave
CCN502.ccn_cache.upstream[3]	PVBusSlave
CCN502.ccn_cache.upstream[40]	PVBusSlave
CCN502.ccn_cache.upstream[41]	PVBusSlave
CCN502.ccn_cache.upstream[42]	PVBusSlave
CCN502.ccn_cache.upstream[43]	PVBusSlave
CCN502.ccn_cache.upstream[44]	PVBusSlave
CCN502.ccn_cache.upstream[45]	PVBusSlave
CCN502.ccn_cache.upstream[46]	PVBusSlave
CCN502.ccn_cache.upstream[47]	PVBusSlave
CCN502.ccn_cache.upstream[4]	PVBusSlave
CCN502.ccn_cache.upstream[5]	PVBusSlave
CCN502.ccn_cache.upstream[6]	PVBusSlave
CCN502.ccn_cache.upstream[7]	PVBusSlave
CCN502.ccn_cache.upstream[8]	PVBusSlave
CCN502.ccn_cache.upstream[9]	PVBusSlave
CCN502.ccn_registers	CCNRegisterSet
CCN502.ccn_registers.bus_slave	PVBusSlave
CCN502.ccn_router	PVBusMapper

This model has the following MTI trace components:

Table 3-854: CCN502 MTI instances

InstanceName	ComponentName
CCN502.bus_slave_ocm	PVBusSlave
CCN502.ccn502_hni_exclusive_monitor_0	PVBusExclusiveMonitor
CCN502.ccn502_hni_exclusive_monitor_0.bus_mapper	PVBusMapper
CCN502.ccn_cache	CCNCache
CCN502.ccn_cache.upstream[0]	PVBusSlave
CCN502.ccn_cache.upstream[10]	PVBusSlave
CCN502.ccn_cache.upstream[11]	PVBusSlave
CCN502.ccn_cache.upstream[12]	PVBusSlave
CCN502.ccn_cache.upstream[13]	PVBusSlave

InstanceName	ComponentName
CCN502.ccn_cache.upstream[14]	PVBusSlave
CCN502.ccn_cache.upstream[15]	PVBusSlave
CCN502.ccn_cache.upstream[16]	PVBusSlave
CCN502.ccn_cache.upstream[17]	PVBusSlave
CCN502.ccn_cache.upstream[18]	PVBusSlave
CCN502.ccn_cache.upstream[19]	PVBusSlave
CCN502.ccn_cache.upstream[1]	PVBusSlave
CCN502.ccn_cache.upstream[20]	PVBusSlave
CCN502.ccn_cache.upstream[21]	PVBusSlave
CCN502.ccn_cache.upstream[22]	PVBusSlave
CCN502.ccn_cache.upstream[23]	PVBusSlave
CCN502.ccn_cache.upstream[24]	PVBusSlave
CCN502.ccn_cache.upstream[25]	PVBusSlave
CCN502.ccn_cache.upstream[26]	PVBusSlave
CCN502.ccn_cache.upstream[27]	PVBusSlave
CCN502.ccn_cache.upstream[28]	PVBusSlave
CCN502.ccn_cache.upstream[29]	PVBusSlave
CCN502.ccn_cache.upstream[2]	PVBusSlave
CCN502.ccn_cache.upstream[30]	PVBusSlave
CCN502.ccn_cache.upstream[31]	PVBusSlave
CCN502.ccn_cache.upstream[32]	PVBusSlave
CCN502.ccn_cache.upstream[33]	PVBusSlave
CCN502.ccn_cache.upstream[34]	PVBusSlave
CCN502.ccn_cache.upstream[35]	PVBusSlave
CCN502.ccn_cache.upstream[36]	PVBusSlave
CCN502.ccn_cache.upstream[37]	PVBusSlave
CCN502.ccn_cache.upstream[38]	PVBusSlave
CCN502.ccn_cache.upstream[39]	PVBusSlave
CCN502.ccn_cache.upstream[3]	PVBusSlave
CCN502.ccn_cache.upstream[40]	PVBusSlave
CCN502.ccn_cache.upstream[41]	PVBusSlave
CCN502.ccn_cache.upstream[42]	PVBusSlave
CCN502.ccn_cache.upstream[43]	PVBusSlave
CCN502.ccn_cache.upstream[44]	PVBusSlave
CCN502.ccn_cache.upstream[45]	PVBusSlave
CCN502.ccn_cache.upstream[46]	PVBusSlave
CCN502.ccn_cache.upstream[47]	PVBusSlave
CCN502.ccn_cache.upstream[4]	PVBusSlave
CCN502.ccn_cache.upstream[5]	PVBusSlave

InstanceName	ComponentName
CCN502.ccn_cache.upstream[6]	PVBusSlave
CCN502.ccn_cache.upstream[7]	PVBusSlave
CCN502.ccn_cache.upstream[8]	PVBusSlave
CCN502.ccn_cache.upstream[9]	PVBusSlave
CCN502.ccn_registers	CCNRegisterSet
CCN502.ccn_registers.bus_slave	PVBusSlave
CCN502.ccn_router	PVBusMapper

CCN502 contains the following CADI targets:

- CCN5XX
- CCNCache

About CCN502

CCN502 can be used for connecting components using the ACE and ACE-Lite interfaces. It has an L3 cache that can provide coherency between up to four fully coherent ACE clusters and nine I/O coherent masters. It can connect up to two memory elements to drive transaction requests. This interconnect can be configured to support six or eight crosspoints, both of which are implemented in the model.

CCN502 has three or five downstream ports, depending on the number of crosspoints:

- Two or four SN-F ports for the memory controller
- One Acelite port (HNI)

The CCN502 parameters are not exposed through CADI, but can be seen in `ccn502.lisa`. Some useful parameters are:

- `cache_state_modelled`
- `cache_size_in_kbytes`
- `systemaddrmap`
- `variant_name`

CCN-502 supports up to 4 SN-Fs. To enable 4 SN-Fs in the model, set the `variant_name` parameter to `ccn502_8xp`, which means the 8XP/4HNF configuration.

The model has the following limitations:

- No support for 3 SN striping.
- If there are multiple SN-Fs, the distribution of addresses to each SN-F is not guaranteed to match the hardware.
- The parameters for the `ccn_cache` subcomponent are not accessible in System Canvas.

Ports for CCN502

Table 3-855: Ports

Name	Protocol	Type	Description
pvbus_m_hni[1]	PVBus	Master	HNI downstream port.
pvbus_m_snf[4]	PVBus	Master	SNF downstream ports.
pvbus_s_rnf[4]	PVBus	Slave	RNF upstream ports.
pvbus_s_rni[9]	PVBus	Slave	RNI upstream ports
reset_in	Signal	Slave	Reset signal.

Parameters for CCN502

acchannelen_rnf

For each upstream port, determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0xf

acchannelen_rni

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

int

Default value

0x1ff

cache_size_in_kbytes

Number of kilo bytes in cache

Type

int

Default value

0x1000

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

ccn_cache.acchannelen_rnf

For each upstream port, determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0xf

ccn_cache.acchannelen_rni

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

int

Default value

0xffff

ccn_cache.cache_size_kb

Number of kilo bytes in cache

Type

int

Default value

0x2000

ccn_cache.cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

ccn_cache.force_on_from_start

Enables snooping on upstream ports from the start of simulation. The CCI will normally start up with snooping disabled, however, using this parameter we allow the model to start with snooping enabled without having to program it. This is only setup at simulation reset and not at signal reset. If the upstreams can ever be held in reset then you **must** connect the reset_state_of_upstream_port_3 and reset_state_of_upstream_port_4 so that it knows when to disable snoops to the upstream systems. Otherwise, the upstream system will receive snoop messages whilst in reset and it will complain that it 'received a snoop request whilst it was in reset'.

Type

bool

Default value

0x0

ccn_cache.number_of_phantom_entries

Number of phantom entries in the cache. Phantom entries are used by certain cache operations to hold temporary data. Usually this should be left at the default value which is safe for all systems containing up to 32 masters.

Type

int

Default value

0x20

ccn_cache.periphbase

Value for PERIPHBASE. Only bits [43:24] are used

Type

int

Default value

0x2c000000

ccn_cache.reentrancy_support

Must be one of: on/off/cacheglobal/env: 'on': hazard checking per cache line (normal mode), 'off': no hazard checking (use only for single master systems), 'cacheglobal': hazard checking globally for cache (not per cache line, testing feature, provokes more hazards than necessary), 'env' (or empty string): take value from FM_REENTRANCY_SUPPORT env var, if this is not set use 'on', default is 'env'

Type

string

Default value

env

ccn_cache.sbas_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

ccn_cache.sbsx_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

enable_logger

Enable PVBUSLogger for downstream ports

Type

bool

Default value

0x0

force_on_from_start

Enables snooping on upstream ports from the start of simulation. The CCN will normally start up with snooping disabled, however, using this parameter we allow the model to start with snooping enabled without having to program it. This is only setup at simulation reset and not at signal reset.

Type

bool

Default value

0x0

number_of_snf

Number of SNF nodes present.

Type

int

Default value

0x2

periphbase

Value for PERIPHBASE. Only bits [43:24] are used

Type

int

Default value

0x2c000000

sbsx_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

systemaddrmap

Bitmap for 20 regions in CCN Interconnect. Every two bits describes the region type for corresponding region.

Type

int

Default value

0x0

variant_name

Can be either CCN502_6XP or CCN502_8XP

Type

string

Default value

CCN502_8XP

3.10.9 CCN504

CCN504 Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-856: IP revisions support

Revision	Quality level
rOp0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CCN504

This model has the following Iris instances:

Table 3-857: CCN504 Iris instances

InstanceName	ComponentName
CCN504	CCN5XX
CCN504.ccn502_hni_exclusive_monitor_0	PVBusExclusiveMonitor
CCN504.ccn502_hni_exclusive_monitor_0.bus_mapper	PVBusMapper
CCN504.ccn_cache	CCNCache
CCN504.ccn_cache.upstream[0]	PVBusSlave
CCN504.ccn_cache.upstream[10]	PVBusSlave
CCN504.ccn_cache.upstream[11]	PVBusSlave
CCN504.ccn_cache.upstream[12]	PVBusSlave
CCN504.ccn_cache.upstream[13]	PVBusSlave
CCN504.ccn_cache.upstream[14]	PVBusSlave
CCN504.ccn_cache.upstream[15]	PVBusSlave
CCN504.ccn_cache.upstream[16]	PVBusSlave
CCN504.ccn_cache.upstream[17]	PVBusSlave

InstanceName	ComponentName
CCN504.ccn_cache.upstream[18]	PVBusSlave
CCN504.ccn_cache.upstream[19]	PVBusSlave
CCN504.ccn_cache.upstream[1]	PVBusSlave
CCN504.ccn_cache.upstream[20]	PVBusSlave
CCN504.ccn_cache.upstream[21]	PVBusSlave
CCN504.ccn_cache.upstream[22]	PVBusSlave
CCN504.ccn_cache.upstream[23]	PVBusSlave
CCN504.ccn_cache.upstream[24]	PVBusSlave
CCN504.ccn_cache.upstream[25]	PVBusSlave
CCN504.ccn_cache.upstream[26]	PVBusSlave
CCN504.ccn_cache.upstream[27]	PVBusSlave
CCN504.ccn_cache.upstream[28]	PVBusSlave
CCN504.ccn_cache.upstream[29]	PVBusSlave
CCN504.ccn_cache.upstream[2]	PVBusSlave
CCN504.ccn_cache.upstream[30]	PVBusSlave
CCN504.ccn_cache.upstream[31]	PVBusSlave
CCN504.ccn_cache.upstream[32]	PVBusSlave
CCN504.ccn_cache.upstream[33]	PVBusSlave
CCN504.ccn_cache.upstream[34]	PVBusSlave
CCN504.ccn_cache.upstream[35]	PVBusSlave
CCN504.ccn_cache.upstream[36]	PVBusSlave
CCN504.ccn_cache.upstream[37]	PVBusSlave
CCN504.ccn_cache.upstream[38]	PVBusSlave
CCN504.ccn_cache.upstream[39]	PVBusSlave
CCN504.ccn_cache.upstream[3]	PVBusSlave
CCN504.ccn_cache.upstream[40]	PVBusSlave
CCN504.ccn_cache.upstream[41]	PVBusSlave
CCN504.ccn_cache.upstream[42]	PVBusSlave
CCN504.ccn_cache.upstream[43]	PVBusSlave
CCN504.ccn_cache.upstream[44]	PVBusSlave
CCN504.ccn_cache.upstream[45]	PVBusSlave
CCN504.ccn_cache.upstream[46]	PVBusSlave
CCN504.ccn_cache.upstream[47]	PVBusSlave
CCN504.ccn_cache.upstream[4]	PVBusSlave
CCN504.ccn_cache.upstream[5]	PVBusSlave
CCN504.ccn_cache.upstream[6]	PVBusSlave
CCN504.ccn_cache.upstream[7]	PVBusSlave
CCN504.ccn_cache.upstream[8]	PVBusSlave
CCN504.ccn_cache.upstream[9]	PVBusSlave

InstanceName	ComponentName
CCN504.ccn_registers	CCNRegisterSet
CCN504.ccn_registers.bus_slave	PVBusSlave
CCN504.ccn_router	PVBusMapper

This model has the following MTI trace components:

Table 3-858: CCN504 MTI instances

InstanceName	ComponentName
CCN504.ccn502_hni_exclusive_monitor_0	PVBusExclusiveMonitor
CCN504.ccn502_hni_exclusive_monitor_0.bus_mapper	PVBusMapper
CCN504.ccn_cache	CCNCache
CCN504.ccn_cache.upstream[0]	PVBusSlave
CCN504.ccn_cache.upstream[10]	PVBusSlave
CCN504.ccn_cache.upstream[11]	PVBusSlave
CCN504.ccn_cache.upstream[12]	PVBusSlave
CCN504.ccn_cache.upstream[13]	PVBusSlave
CCN504.ccn_cache.upstream[14]	PVBusSlave
CCN504.ccn_cache.upstream[15]	PVBusSlave
CCN504.ccn_cache.upstream[16]	PVBusSlave
CCN504.ccn_cache.upstream[17]	PVBusSlave
CCN504.ccn_cache.upstream[18]	PVBusSlave
CCN504.ccn_cache.upstream[19]	PVBusSlave
CCN504.ccn_cache.upstream[1]	PVBusSlave
CCN504.ccn_cache.upstream[20]	PVBusSlave
CCN504.ccn_cache.upstream[21]	PVBusSlave
CCN504.ccn_cache.upstream[22]	PVBusSlave
CCN504.ccn_cache.upstream[23]	PVBusSlave
CCN504.ccn_cache.upstream[24]	PVBusSlave
CCN504.ccn_cache.upstream[25]	PVBusSlave
CCN504.ccn_cache.upstream[26]	PVBusSlave
CCN504.ccn_cache.upstream[27]	PVBusSlave
CCN504.ccn_cache.upstream[28]	PVBusSlave
CCN504.ccn_cache.upstream[29]	PVBusSlave
CCN504.ccn_cache.upstream[2]	PVBusSlave
CCN504.ccn_cache.upstream[30]	PVBusSlave
CCN504.ccn_cache.upstream[31]	PVBusSlave
CCN504.ccn_cache.upstream[32]	PVBusSlave
CCN504.ccn_cache.upstream[33]	PVBusSlave
CCN504.ccn_cache.upstream[34]	PVBusSlave
CCN504.ccn_cache.upstream[35]	PVBusSlave

InstanceName	ComponentName
CCN504.ccn_cache.upstream[36]	PVBusSlave
CCN504.ccn_cache.upstream[37]	PVBusSlave
CCN504.ccn_cache.upstream[38]	PVBusSlave
CCN504.ccn_cache.upstream[39]	PVBusSlave
CCN504.ccn_cache.upstream[3]	PVBusSlave
CCN504.ccn_cache.upstream[40]	PVBusSlave
CCN504.ccn_cache.upstream[41]	PVBusSlave
CCN504.ccn_cache.upstream[42]	PVBusSlave
CCN504.ccn_cache.upstream[43]	PVBusSlave
CCN504.ccn_cache.upstream[44]	PVBusSlave
CCN504.ccn_cache.upstream[45]	PVBusSlave
CCN504.ccn_cache.upstream[46]	PVBusSlave
CCN504.ccn_cache.upstream[47]	PVBusSlave
CCN504.ccn_cache.upstream[4]	PVBusSlave
CCN504.ccn_cache.upstream[5]	PVBusSlave
CCN504.ccn_cache.upstream[6]	PVBusSlave
CCN504.ccn_cache.upstream[7]	PVBusSlave
CCN504.ccn_cache.upstream[8]	PVBusSlave
CCN504.ccn_cache.upstream[9]	PVBusSlave
CCN504.ccn_registers	CCNRegisterSet
CCN504.ccn_registers.bus_slave	PVBusSlave
CCN504.ccn_router	PVBusMapper

CCN504 contains the following CADI targets:

- CCN5XX
- CCNCache

About CCN504

CCN504 has three downstream ports:

- Two SNF ports for the memory controller
- One Acelite port (HNI)



The parameters for the `ccncache` subcomponent are not accessible in System Canvas.

Ports for CCN504

Table 3-859: Ports

Name	Protocol	Type	Description
pvbus_m_hni[1]	PVBus	Master	HNI downstream port.
pvbus_m_snf[2]	PVBus	Master	SNF downstream ports.
pvbus_s_rnf[4]	PVBus	Slave	RNF upstream ports.
pvbus_s_rni[18]	PVBus	Slave	RNI upstream ports.
reset_in	Signal	Slave	Reset signal.

Parameters for CCN504

acchannelen_rnf

For each upstream port, determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0xf

acchannelen_rni

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

int

Default value

0x3ffff

cache_size_in_mbytes

Number of mega bytes in cache

Type

int

Default value

0x8

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

ccn_cache.acchannelen_rnf

For each upstream port, determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0xf

ccn_cache.acchannelen_rni

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

int

Default value

0xffff

ccn_cache.cache_size_kb

Number of kilo bytes in cache

Type

int

Default value

0x2000

ccn_cache.cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

ccn_cache.force_on_from_start

Enables snooping on upstream ports from the start of simulation. The CCI will normally start up with snooping disabled, however, using this parameter we allow the model to start with snooping enabled without having to program it. This is only setup at simulation reset and not at signal reset. If the upstreams can ever be held in reset then you **must** connect the reset_state_of_upstream_port_3 and reset_state_of_upstream_port_4 so that it knows when to disable snoops to the upstream systems. Otherwise, the upstream system will receive snoop messages whilst in reset and it will complain that it 'received a snoop request whilst it was in reset'.

Type

bool

Default value

0x0

ccn_cache.number_of_phantom_entries

Number of phantom entries in the cache. Phantom entries are used by certain cache operations to hold temporary data. Usually this should be left at the default value which is safe for all systems containing up to 32 masters.

Type

int

Default value

0x20

ccn_cache.periphbase

Value for PERIPHBASE. Only bits [43:24] are used

Type

int

Default value

0x2c000000

ccn_cache.reentrancy_support

Must be one of: on/off/cacheglobal/env: 'on': hazard checking per cache line (normal mode), 'off': no hazard checking (use only for single master systems), 'cacheglobal': hazard checking globally for cache (not per cache line, testing feature, provokes more hazards than necessary), 'env' (or empty string): take value from FM_REENTRANCY_SUPPORT env var, if this is not set use 'on', default is 'env'

Type

string

Default value

env

ccn_cache.sbas_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

ccn_cache.sbsx_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

disable_hni_cacheable_error

Disable sending error response when HNI receives cacheable access.

Type

bool

Default value

0x0

enable_logger

Enable PVBUSLoggers for the downstream ports in the CCN model.

Type

bool

Default value

0x0

force_on_from_start

Enables snooping on upstream ports from the start of simulation. The CCN will normally start up with snooping disabled, however, using this parameter we allow the model to start with snooping enabled without having to program it. This is only setup at simulation reset and not at signal reset.

Type

bool

Default value

0x0

number_of_snf

Number of SNF nodes present.

Type

int

Default value

0x2

periphbase

Value for PERIPHBASE. Only bits [43:24] are used

Type

int

Default value

0x2c000000

sbas_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

sbsx_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

systemaddrmap

Bitmap for 20 regions in CCN Interconnect. Every two bits describes the region type for corresponding region.

Type

int

Default value

0x0

3.10.10 CCN508

CCN508 Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-860: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CCN508

This model has the following Iris instances:

Table 3-861: CCN508 Iris instances

InstanceName	ComponentName
CCN508	CCN5XX
CCN508.ccn502_hni_exclusive_monitor_0	PVBusExclusiveMonitor
CCN508.ccn502_hni_exclusive_monitor_0.bus_mapper	PVBusMapper
CCN508.ccn502_hni_exclusive_monitor_1	PVBusExclusiveMonitor
CCN508.ccn502_hni_exclusive_monitor_1.bus_mapper	PVBusMapper

InstanceName	ComponentName
CCN508.ccn_cache	CCNCache
CCN508.ccn_cache.upstream[0]	PVBusSlave
CCN508.ccn_cache.upstream[10]	PVBusSlave
CCN508.ccn_cache.upstream[11]	PVBusSlave
CCN508.ccn_cache.upstream[12]	PVBusSlave
CCN508.ccn_cache.upstream[13]	PVBusSlave
CCN508.ccn_cache.upstream[14]	PVBusSlave
CCN508.ccn_cache.upstream[15]	PVBusSlave
CCN508.ccn_cache.upstream[16]	PVBusSlave
CCN508.ccn_cache.upstream[17]	PVBusSlave
CCN508.ccn_cache.upstream[18]	PVBusSlave
CCN508.ccn_cache.upstream[19]	PVBusSlave
CCN508.ccn_cache.upstream[1]	PVBusSlave
CCN508.ccn_cache.upstream[20]	PVBusSlave
CCN508.ccn_cache.upstream[21]	PVBusSlave
CCN508.ccn_cache.upstream[22]	PVBusSlave
CCN508.ccn_cache.upstream[23]	PVBusSlave
CCN508.ccn_cache.upstream[24]	PVBusSlave
CCN508.ccn_cache.upstream[25]	PVBusSlave
CCN508.ccn_cache.upstream[26]	PVBusSlave
CCN508.ccn_cache.upstream[27]	PVBusSlave
CCN508.ccn_cache.upstream[28]	PVBusSlave
CCN508.ccn_cache.upstream[29]	PVBusSlave
CCN508.ccn_cache.upstream[2]	PVBusSlave
CCN508.ccn_cache.upstream[30]	PVBusSlave
CCN508.ccn_cache.upstream[31]	PVBusSlave
CCN508.ccn_cache.upstream[32]	PVBusSlave
CCN508.ccn_cache.upstream[33]	PVBusSlave
CCN508.ccn_cache.upstream[34]	PVBusSlave
CCN508.ccn_cache.upstream[35]	PVBusSlave
CCN508.ccn_cache.upstream[36]	PVBusSlave
CCN508.ccn_cache.upstream[37]	PVBusSlave
CCN508.ccn_cache.upstream[38]	PVBusSlave
CCN508.ccn_cache.upstream[39]	PVBusSlave
CCN508.ccn_cache.upstream[3]	PVBusSlave
CCN508.ccn_cache.upstream[40]	PVBusSlave
CCN508.ccn_cache.upstream[41]	PVBusSlave
CCN508.ccn_cache.upstream[42]	PVBusSlave
CCN508.ccn_cache.upstream[43]	PVBusSlave

InstanceName	ComponentName
CCN508.ccn_cache.upstream[44]	PVBusSlave
CCN508.ccn_cache.upstream[45]	PVBusSlave
CCN508.ccn_cache.upstream[46]	PVBusSlave
CCN508.ccn_cache.upstream[47]	PVBusSlave
CCN508.ccn_cache.upstream[4]	PVBusSlave
CCN508.ccn_cache.upstream[5]	PVBusSlave
CCN508.ccn_cache.upstream[6]	PVBusSlave
CCN508.ccn_cache.upstream[7]	PVBusSlave
CCN508.ccn_cache.upstream[8]	PVBusSlave
CCN508.ccn_cache.upstream[9]	PVBusSlave
CCN508.ccn_registers	CCNRegisterSet
CCN508.ccn_registers.bus_slave	PVBusSlave
CCN508.ccn_router	PVBusMapper

This model has the following MTI trace components:

Table 3-862: CCN508 MTI instances

InstanceName	ComponentName
CCN508.ccn502_hni_exclusive_monitor_0	PVBusExclusiveMonitor
CCN508.ccn502_hni_exclusive_monitor_0.bus_mapper	PVBusMapper
CCN508.ccn502_hni_exclusive_monitor_1	PVBusExclusiveMonitor
CCN508.ccn502_hni_exclusive_monitor_1.bus_mapper	PVBusMapper
CCN508.ccn_cache	CCNCache
CCN508.ccn_cache.upstream[0]	PVBusSlave
CCN508.ccn_cache.upstream[10]	PVBusSlave
CCN508.ccn_cache.upstream[11]	PVBusSlave
CCN508.ccn_cache.upstream[12]	PVBusSlave
CCN508.ccn_cache.upstream[13]	PVBusSlave
CCN508.ccn_cache.upstream[14]	PVBusSlave
CCN508.ccn_cache.upstream[15]	PVBusSlave
CCN508.ccn_cache.upstream[16]	PVBusSlave
CCN508.ccn_cache.upstream[17]	PVBusSlave
CCN508.ccn_cache.upstream[18]	PVBusSlave
CCN508.ccn_cache.upstream[19]	PVBusSlave
CCN508.ccn_cache.upstream[1]	PVBusSlave
CCN508.ccn_cache.upstream[20]	PVBusSlave
CCN508.ccn_cache.upstream[21]	PVBusSlave
CCN508.ccn_cache.upstream[22]	PVBusSlave
CCN508.ccn_cache.upstream[23]	PVBusSlave
CCN508.ccn_cache.upstream[24]	PVBusSlave

InstanceName	ComponentName
CCN508.ccn_cache.upstream[25]	PVBusSlave
CCN508.ccn_cache.upstream[26]	PVBusSlave
CCN508.ccn_cache.upstream[27]	PVBusSlave
CCN508.ccn_cache.upstream[28]	PVBusSlave
CCN508.ccn_cache.upstream[29]	PVBusSlave
CCN508.ccn_cache.upstream[2]	PVBusSlave
CCN508.ccn_cache.upstream[30]	PVBusSlave
CCN508.ccn_cache.upstream[31]	PVBusSlave
CCN508.ccn_cache.upstream[32]	PVBusSlave
CCN508.ccn_cache.upstream[33]	PVBusSlave
CCN508.ccn_cache.upstream[34]	PVBusSlave
CCN508.ccn_cache.upstream[35]	PVBusSlave
CCN508.ccn_cache.upstream[36]	PVBusSlave
CCN508.ccn_cache.upstream[37]	PVBusSlave
CCN508.ccn_cache.upstream[38]	PVBusSlave
CCN508.ccn_cache.upstream[39]	PVBusSlave
CCN508.ccn_cache.upstream[3]	PVBusSlave
CCN508.ccn_cache.upstream[40]	PVBusSlave
CCN508.ccn_cache.upstream[41]	PVBusSlave
CCN508.ccn_cache.upstream[42]	PVBusSlave
CCN508.ccn_cache.upstream[43]	PVBusSlave
CCN508.ccn_cache.upstream[44]	PVBusSlave
CCN508.ccn_cache.upstream[45]	PVBusSlave
CCN508.ccn_cache.upstream[46]	PVBusSlave
CCN508.ccn_cache.upstream[47]	PVBusSlave
CCN508.ccn_cache.upstream[4]	PVBusSlave
CCN508.ccn_cache.upstream[5]	PVBusSlave
CCN508.ccn_cache.upstream[6]	PVBusSlave
CCN508.ccn_cache.upstream[7]	PVBusSlave
CCN508.ccn_cache.upstream[8]	PVBusSlave
CCN508.ccn_cache.upstream[9]	PVBusSlave
CCN508.ccn_registers	CCNRegisterSet
CCN508.ccn_registers.bus_slave	PVBusSlave
CCN508.ccn_router	PVBusMapper

CCN508 contains the following CADI targets:

- CCN5XX
- CCNCache

About CCN508

CCN508 has six downstream ports:

- Four SNF ports for the memory controller.
- Two Acelite ports (HNI).



The parameters for the `ccncache` subcomponent are not accessible in System Canvas.

Ports for CCN508

Table 3-863: Ports

Name	Protocol	Type	Description
<code>pvbus_m_hni[2]</code>	PVBus	Master	HNI downstream ports.
<code>pvbus_m_snf[4]</code>	PVBus	Master	SNF downstream ports.
<code>pvbus_s_rnf[8]</code>	PVBus	Slave	RNF upstream ports.
<code>pvbus_s_rni[24]</code>	PVBus	Slave	RNI upstream ports.
<code>reset_in</code>	Signal	Slave	Reset signal.

Parameters for CCN508

`acchannelen_rnf`

For each upstream port, determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0xf

`acchannelen_rni`

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

int

Default value

0x3ffff

`cache_size_in_mbytes`

Number of mega bytes in cache

Type

int

Default value

0x8

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

ccn_cache.acchannelen_rnf

For each upstream port, determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0xf

ccn_cache.acchannelen_rni

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

int

Default value

0xffff

ccn_cache.cache_size_kb

Number of kilo bytes in cache

Type

int

Default value

0x2000

ccn_cache.cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

ccn_cache.force_on_from_start

Enables snooping on upstream ports from the start of simulation. The CCI will normally start up with snooping disabled, however, using this parameter we allow the model to start with snooping enabled without having to program it. This is only setup at simulation reset and not at signal reset. If the upstreams can ever be held in reset then you **must** connect the reset_state_of_upstream_port_3 and reset_state_of_upstream_port_4 so that it knows when to disable snoops to the upstream systems. Otherwise, the upstream system will receive

snoop messages whilst in reset and it will complain that it 'received a snoop request whilst it was in reset'.

Type

bool

Default value

0x0

ccn_cache.number_of_phantom_entries

Number of phantom entries in the cache. Phantom entries are used by certain cache operations to hold temporary data. Usually this should be left at the default value which is safe for all systems containing up to 32 masters.

Type

int

Default value

0x20

ccn_cache.periphbase

Value for PERIPHBASE. Only bits [43:24] are used

Type

int

Default value

0x2c000000

ccn_cache.reentrancy_support

Must be one of: on/off/cacheglobal/env: 'on': hazard checking per cache line (normal mode), 'off': no hazard checking (use only for single master systems), 'cacheglobal': hazard checking globally for cache (not per cache line, testing feature, provokes more hazards than necessary), 'env' (or empty string): take value from FM_REENTRANCY_SUPPORT env var, if this is not set use 'on', default is 'env'

Type

string

Default value

env

ccn_cache.sbas_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

ccn_cache.sbsx_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

disable_hni_cacheable_error

Disable sending error response when HNI receives cacheable access.

Type

bool

Default value

0x0

enable_logger

Enable PVBUSLoggers for the downstream ports in the CCN model.

Type

bool

Default value

0x0

force_on_from_start

Enables snooping on upstream ports from the start of simulation. The CCN will normally start up with snooping disabled, however, using this parameter we allow the model to start with snooping enabled without having to program it. This is only setup at simulation reset and not at signal reset.

Type

bool

Default value

0x0

number_of_snf

Number of SNF nodes present.

Type

int

Default value

0x2

periphbase

Value for PERIPHBASE. Only bits [43:24] are used

Type

int

Default value

0x2c000000

sbas_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

sbsx_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

systemaddrmap

Bitmap for 20 regions in CCN Interconnect. Every two bits describes the region type for corresponding region.

Type

int

Default value

0x0

3.10.11 CCN512

CCN512 Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-864: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CCN512

This model has the following Iris instances:

Table 3-865: CCN512 Iris instances

InstanceName	ComponentName
CCN512	CCN5XX
CCN512.bus_slave_ocm	PVBusSlave
CCN512.ccn502_hni_exclusive_monitor_0	PVBusExclusiveMonitor
CCN512.ccn502_hni_exclusive_monitor_0.bus_mapper	PVBusMapper
CCN512.ccn502_hni_exclusive_monitor_1	PVBusExclusiveMonitor
CCN512.ccn502_hni_exclusive_monitor_1.bus_mapper	PVBusMapper
CCN512.ccn_cache	CCNCache
CCN512.ccn_cache.upstream[0]	PVBusSlave
CCN512.ccn_cache.upstream[10]	PVBusSlave
CCN512.ccn_cache.upstream[11]	PVBusSlave
CCN512.ccn_cache.upstream[12]	PVBusSlave
CCN512.ccn_cache.upstream[13]	PVBusSlave
CCN512.ccn_cache.upstream[14]	PVBusSlave
CCN512.ccn_cache.upstream[15]	PVBusSlave
CCN512.ccn_cache.upstream[16]	PVBusSlave
CCN512.ccn_cache.upstream[17]	PVBusSlave
CCN512.ccn_cache.upstream[18]	PVBusSlave
CCN512.ccn_cache.upstream[19]	PVBusSlave
CCN512.ccn_cache.upstream[1]	PVBusSlave
CCN512.ccn_cache.upstream[20]	PVBusSlave
CCN512.ccn_cache.upstream[21]	PVBusSlave
CCN512.ccn_cache.upstream[22]	PVBusSlave
CCN512.ccn_cache.upstream[23]	PVBusSlave
CCN512.ccn_cache.upstream[24]	PVBusSlave
CCN512.ccn_cache.upstream[25]	PVBusSlave
CCN512.ccn_cache.upstream[26]	PVBusSlave
CCN512.ccn_cache.upstream[27]	PVBusSlave
CCN512.ccn_cache.upstream[28]	PVBusSlave
CCN512.ccn_cache.upstream[29]	PVBusSlave
CCN512.ccn_cache.upstream[2]	PVBusSlave
CCN512.ccn_cache.upstream[30]	PVBusSlave
CCN512.ccn_cache.upstream[31]	PVBusSlave
CCN512.ccn_cache.upstream[32]	PVBusSlave
CCN512.ccn_cache.upstream[33]	PVBusSlave
CCN512.ccn_cache.upstream[34]	PVBusSlave
CCN512.ccn_cache.upstream[35]	PVBusSlave
CCN512.ccn_cache.upstream[36]	PVBusSlave
CCN512.ccn_cache.upstream[37]	PVBusSlave

InstanceName	ComponentName
CCN512.ccn_cache.upstream[38]	PVBusSlave
CCN512.ccn_cache.upstream[39]	PVBusSlave
CCN512.ccn_cache.upstream[3]	PVBusSlave
CCN512.ccn_cache.upstream[40]	PVBusSlave
CCN512.ccn_cache.upstream[41]	PVBusSlave
CCN512.ccn_cache.upstream[42]	PVBusSlave
CCN512.ccn_cache.upstream[43]	PVBusSlave
CCN512.ccn_cache.upstream[44]	PVBusSlave
CCN512.ccn_cache.upstream[45]	PVBusSlave
CCN512.ccn_cache.upstream[46]	PVBusSlave
CCN512.ccn_cache.upstream[47]	PVBusSlave
CCN512.ccn_cache.upstream[4]	PVBusSlave
CCN512.ccn_cache.upstream[5]	PVBusSlave
CCN512.ccn_cache.upstream[6]	PVBusSlave
CCN512.ccn_cache.upstream[7]	PVBusSlave
CCN512.ccn_cache.upstream[8]	PVBusSlave
CCN512.ccn_cache.upstream[9]	PVBusSlave
CCN512.ccn_registers	CCNRegisterSet
CCN512.ccn_registers.bus_slave	PVBusSlave
CCN512.ccn_router	PVBusMapper

This model has the following MTI trace components:

Table 3-866: CCN512 MTI instances

InstanceName	ComponentName
CCN512.bus_slave_ocm	PVBusSlave
CCN512.ccn502_hni_exclusive_monitor_0	PVBusExclusiveMonitor
CCN512.ccn502_hni_exclusive_monitor_0.bus_mapper	PVBusMapper
CCN512.ccn502_hni_exclusive_monitor_1	PVBusExclusiveMonitor
CCN512.ccn502_hni_exclusive_monitor_1.bus_mapper	PVBusMapper
CCN512.ccn_cache	CCNCache
CCN512.ccn_cache.upstream[0]	PVBusSlave
CCN512.ccn_cache.upstream[10]	PVBusSlave
CCN512.ccn_cache.upstream[11]	PVBusSlave
CCN512.ccn_cache.upstream[12]	PVBusSlave
CCN512.ccn_cache.upstream[13]	PVBusSlave
CCN512.ccn_cache.upstream[14]	PVBusSlave
CCN512.ccn_cache.upstream[15]	PVBusSlave
CCN512.ccn_cache.upstream[16]	PVBusSlave
CCN512.ccn_cache.upstream[17]	PVBusSlave

InstanceName	ComponentName
CCN512.ccn_cache.upstream[18]	PVBusSlave
CCN512.ccn_cache.upstream[19]	PVBusSlave
CCN512.ccn_cache.upstream[1]	PVBusSlave
CCN512.ccn_cache.upstream[20]	PVBusSlave
CCN512.ccn_cache.upstream[21]	PVBusSlave
CCN512.ccn_cache.upstream[22]	PVBusSlave
CCN512.ccn_cache.upstream[23]	PVBusSlave
CCN512.ccn_cache.upstream[24]	PVBusSlave
CCN512.ccn_cache.upstream[25]	PVBusSlave
CCN512.ccn_cache.upstream[26]	PVBusSlave
CCN512.ccn_cache.upstream[27]	PVBusSlave
CCN512.ccn_cache.upstream[28]	PVBusSlave
CCN512.ccn_cache.upstream[29]	PVBusSlave
CCN512.ccn_cache.upstream[2]	PVBusSlave
CCN512.ccn_cache.upstream[30]	PVBusSlave
CCN512.ccn_cache.upstream[31]	PVBusSlave
CCN512.ccn_cache.upstream[32]	PVBusSlave
CCN512.ccn_cache.upstream[33]	PVBusSlave
CCN512.ccn_cache.upstream[34]	PVBusSlave
CCN512.ccn_cache.upstream[35]	PVBusSlave
CCN512.ccn_cache.upstream[36]	PVBusSlave
CCN512.ccn_cache.upstream[37]	PVBusSlave
CCN512.ccn_cache.upstream[38]	PVBusSlave
CCN512.ccn_cache.upstream[39]	PVBusSlave
CCN512.ccn_cache.upstream[3]	PVBusSlave
CCN512.ccn_cache.upstream[40]	PVBusSlave
CCN512.ccn_cache.upstream[41]	PVBusSlave
CCN512.ccn_cache.upstream[42]	PVBusSlave
CCN512.ccn_cache.upstream[43]	PVBusSlave
CCN512.ccn_cache.upstream[44]	PVBusSlave
CCN512.ccn_cache.upstream[45]	PVBusSlave
CCN512.ccn_cache.upstream[46]	PVBusSlave
CCN512.ccn_cache.upstream[47]	PVBusSlave
CCN512.ccn_cache.upstream[4]	PVBusSlave
CCN512.ccn_cache.upstream[5]	PVBusSlave
CCN512.ccn_cache.upstream[6]	PVBusSlave
CCN512.ccn_cache.upstream[7]	PVBusSlave
CCN512.ccn_cache.upstream[8]	PVBusSlave
CCN512.ccn_cache.upstream[9]	PVBusSlave

InstanceName	ComponentName
CCN512.ccn_registers	CCNRegisterSet
CCN512.ccn_registers.bus_slave	PVBusSlave
CCN512.ccn_router	PVBusMapper

CCN512 contains the following CADI targets:

- CCN5XX
- CCNCache

About CCN512

CCN512 has an L3 cache that can provide coherency between up to 12 fully coherent ACE clusters and 24 I/O coherent masters. It can connect up to four memory elements to drive transaction requests.

CCN512 has six downstream ports:

- Four SNF ports for the memory controller
- Two Acelite ports (HNI)

Some useful parameters are:

- `cache_state_modelled`
- `cache_size_in_mbytes`
- `systemaddrmap`

The model has the following limitations:

- No support for 3 SN striping.
- If there are multiple SN-Fs, the distribution of addresses to each SN-F is not guaranteed to match the hardware.
- The parameters for the `ccnCache` subcomponent are not accessible in System Canvas.

Ports for CCN512

Table 3-867: Ports

Name	Protocol	Type	Description
<code>pdbus_m_hni[2]</code>	PVBus	Master	HNI downstream ports.
<code>pdbus_m_snf[4]</code>	PVBus	Master	SNF downstream ports.
<code>pdbus_s_rnf[12]</code>	PVBus	Slave	RNF upstream ports.
<code>pdbus_s_rni[24]</code>	PVBus	Slave	RNI upstream ports.
<code>reset_in</code>	Signal	Slave	Reset signal.

Parameters for CCN512

`acchannelen_rnf`

For each upstream port, determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0xfff

acchannelen_rni

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

int

Default value

0xffffffff

cache_size_in_mbytes

Number of mega bytes in cache

Type

int

Default value

0x8

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

ccn_cache.acchannelen_rnf

For each upstream port, determine if it is enabled or not w.r.t. snoop requests.

Type

int

Default value

0xf

ccn_cache.acchannelen_rni

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

int

Default value

0xfff

ccn_cache.cache_size_kb

Number of kilo bytes in cache

Type

int

Default value

0x2000

ccn_cache.cache_state_modelled

Model the cache state.

Type

bool

Default value

0x1

ccn_cache.force_on_from_start

Enables snooping on upstream ports from the start of simulation. The CCI will normally start up with snooping disabled, however, using this parameter we allow the model to start with snooping enabled without having to program it. This is only setup at simulation reset and not at signal reset. If the upstreams can ever be held in reset then you **must** connect the reset_state_of_upstream_port_3 and reset_state_of_upstream_port_4 so that it knows when to disable snoops to the upstream systems. Otherwise, the upstream system will receive snoop messages whilst in reset and it will complain that it 'received a snoop request whilst it was in reset'.

Type

bool

Default value

0x0

ccn_cache.number_of_phantom_entries

Number of phantom entries in the cache. Phantom entries are used by certain cache operations to hold temporary data. Usually this should be left at the default value which is safe for all systems containing up to 32 masters.

Type

int

Default value

0x20

ccn_cache.periphbase

Value for PERIPHBASE. Only bits [43:24] are used

Type

int

Default value

0x2c000000

ccn_cache.reentrancy_support

Must be one of: on/off/cacheglobal/env: 'on': hazard checking per cache line (normal mode), 'off': no hazard checking (use only for single master systems), 'cacheglobal': hazard checking globally for cache (not per cache line, testing feature, provokes more hazards than necessary), 'env' (or empty string): take value from FM_REENTRANCY_SUPPORT env var, if this is not set use 'on', default is 'env'

Type

string

Default value

env

ccn_cache.sbas_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

ccn_cache.sbsx_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

enable_logger

Enable PVBUSLogger for downstream ports

Type

bool

Default value

0x0

force_on_from_start

Enables snooping on upstream ports from the start of simulation. The CCN will normally start up with snooping disabled, however, using this parameter we allow the model to start with snooping enabled without having to program it. This is only setup at simulation reset and not at signal reset.

Type

bool

Default value

0x0

number_of_snf

Number of SNF nodes present.

Type

int

Default value

0x2

periphbase

Value for PERIPHBASE. Only bits [43:24] are used

Type

int

Default value

0x2c000000

sbsx_bridge_present

For each upstream port, determine if it is enabled or not w.r.t. dvm requests.

Type

bool

Default value

0x1

systemaddrmap

Bitmap for 20 regions in CCN Interconnect. Every two bits describes the region type for corresponding region.

Type

int

Default value

0x0

3.10.12 CI700

CI700 Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-868: IP revisions support

Revision	Quality level
r2p0	Full support

Revision	Quality level
r1p0	Full support
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CI700

This model has the following Iris instances:

Table 3-869: CI700 Iris instances

InstanceName	ComponentName
CI700	CI700
CI700.bus_slave_ocm_NS	PVBusSlave
CI700.bus_slave_ocm_S	PVBusSlave
CI700.ci700_tag_cache	CMN_TAG_CACHE
CI700.ci700_tag_cache.metadata_controller0	ExportTest.CI700.ci700_tag_cachemetadata_controller0
CI700.ci700_tag_cache.metadata_controller0.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller1	ExportTest.CI700.ci700_tag_cachemetadata_controller1
CI700.ci700_tag_cache.metadata_controller1.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller10	ExportTest.CI700.ci700_tag_cachemetadata_controller10
CI700.ci700_tag_cache.metadata_controller10.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller100	ExportTest.CI700.ci700_tag_cachemetadata_controller100
CI700.ci700_tag_cache.metadata_controller100.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller101	ExportTest.CI700.ci700_tag_cachemetadata_controller101
CI700.ci700_tag_cache.metadata_controller101.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller102	ExportTest.CI700.ci700_tag_cachemetadata_controller102
CI700.ci700_tag_cache.metadata_controller102.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller103	ExportTest.CI700.ci700_tag_cachemetadata_controller103
CI700.ci700_tag_cache.metadata_controller103.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller104	ExportTest.CI700.ci700_tag_cachemetadata_controller104
CI700.ci700_tag_cache.metadata_controller104.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller105	ExportTest.CI700.ci700_tag_cachemetadata_controller105
CI700.ci700_tag_cache.metadata_controller105.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller106	ExportTest.CI700.ci700_tag_cachemetadata_controller106

InstanceName	ComponentName
CI700.ci700_tag_cache. metadata_controller106.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller107	ExportTest.CI700.ci700_tag_cachemetadata_controller107
CI700.ci700_tag_cache. metadata_controller107.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller108	ExportTest.CI700.ci700_tag_cachemetadata_controller108
CI700.ci700_tag_cache. metadata_controller108.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller109	ExportTest.CI700.ci700_tag_cachemetadata_controller109
CI700.ci700_tag_cache. metadata_controller109.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller11	ExportTest.CI700.ci700_tag_cachemetadata_controller11
CI700.ci700_tag_cache. metadata_controller11.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller110	ExportTest.CI700.ci700_tag_cachemetadata_controller110
CI700.ci700_tag_cache. metadata_controller110.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller111	ExportTest.CI700.ci700_tag_cachemetadata_controller111
CI700.ci700_tag_cache. metadata_controller111.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller112	ExportTest.CI700.ci700_tag_cachemetadata_controller112
CI700.ci700_tag_cache. metadata_controller112.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller113	ExportTest.CI700.ci700_tag_cachemetadata_controller113
CI700.ci700_tag_cache. metadata_controller113.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller114	ExportTest.CI700.ci700_tag_cachemetadata_controller114
CI700.ci700_tag_cache. metadata_controller114.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller115	ExportTest.CI700.ci700_tag_cachemetadata_controller115
CI700.ci700_tag_cache. metadata_controller115.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller116	ExportTest.CI700.ci700_tag_cachemetadata_controller116
CI700.ci700_tag_cache. metadata_controller116.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller117	ExportTest.CI700.ci700_tag_cachemetadata_controller117
CI700.ci700_tag_cache. metadata_controller117.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller118	ExportTest.CI700.ci700_tag_cachemetadata_controller118
CI700.ci700_tag_cache. metadata_controller118.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller119	ExportTest.CI700.ci700_tag_cachemetadata_controller119
CI700.ci700_tag_cache. metadata_controller119.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.metadata_controller12	ExportTest.CI700.ci700_tag_cachemetadata_controller12
CI700.ci700_tag_cache. metadata_controller12.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller120	ExportTest.CI700.ci700_tag_cachemetadata_controller120
CI700.ci700_tag_cache. metadata_controller120.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller121	ExportTest.CI700.ci700_tag_cachemetadata_controller121
CI700.ci700_tag_cache. metadata_controller121.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller122	ExportTest.CI700.ci700_tag_cachemetadata_controller122
CI700.ci700_tag_cache. metadata_controller122.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller123	ExportTest.CI700.ci700_tag_cachemetadata_controller123
CI700.ci700_tag_cache. metadata_controller123.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller124	ExportTest.CI700.ci700_tag_cachemetadata_controller124
CI700.ci700_tag_cache. metadata_controller124.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller125	ExportTest.CI700.ci700_tag_cachemetadata_controller125
CI700.ci700_tag_cache. metadata_controller125.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller126	ExportTest.CI700.ci700_tag_cachemetadata_controller126
CI700.ci700_tag_cache. metadata_controller126.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller127	ExportTest.CI700.ci700_tag_cachemetadata_controller127
CI700.ci700_tag_cache. metadata_controller127.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller13	ExportTest.CI700.ci700_tag_cachemetadata_controller13
CI700.ci700_tag_cache. metadata_controller13.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller14	ExportTest.CI700.ci700_tag_cachemetadata_controller14
CI700.ci700_tag_cache. metadata_controller14.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller15	ExportTest.CI700.ci700_tag_cachemetadata_controller15
CI700.ci700_tag_cache. metadata_controller15.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller16	ExportTest.CI700.ci700_tag_cachemetadata_controller16
CI700.ci700_tag_cache. metadata_controller16.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller17	ExportTest.CI700.ci700_tag_cachemetadata_controller17
CI700.ci700_tag_cache. metadata_controller17.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller18	ExportTest.CI700.ci700_tag_cachemetadata_controller18

InstanceName	ComponentName
CI700.ci700_tag_cache. metadata_controller18.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller19	ExportTest.CI700.ci700_tag_cachemetadata_controller19
CI700.ci700_tag_cache. metadata_controller19.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller2	ExportTest.CI700.ci700_tag_cachemetadata_controller2
CI700.ci700_tag_cache. metadata_controller2.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller20	ExportTest.CI700.ci700_tag_cachemetadata_controller20
CI700.ci700_tag_cache. metadata_controller20.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller21	ExportTest.CI700.ci700_tag_cachemetadata_controller21
CI700.ci700_tag_cache. metadata_controller21.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller22	ExportTest.CI700.ci700_tag_cachemetadata_controller22
CI700.ci700_tag_cache. metadata_controller22.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller23	ExportTest.CI700.ci700_tag_cachemetadata_controller23
CI700.ci700_tag_cache. metadata_controller23.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller24	ExportTest.CI700.ci700_tag_cachemetadata_controller24
CI700.ci700_tag_cache. metadata_controller24.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller25	ExportTest.CI700.ci700_tag_cachemetadata_controller25
CI700.ci700_tag_cache. metadata_controller25.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller26	ExportTest.CI700.ci700_tag_cachemetadata_controller26
CI700.ci700_tag_cache. metadata_controller26.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller27	ExportTest.CI700.ci700_tag_cachemetadata_controller27
CI700.ci700_tag_cache. metadata_controller27.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller28	ExportTest.CI700.ci700_tag_cachemetadata_controller28
CI700.ci700_tag_cache. metadata_controller28.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller29	ExportTest.CI700.ci700_tag_cachemetadata_controller29
CI700.ci700_tag_cache. metadata_controller29.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller3	ExportTest.CI700.ci700_tag_cachemetadata_controller3
CI700.ci700_tag_cache. metadata_controller3.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller30	ExportTest.CI700.ci700_tag_cachemetadata_controller30
CI700.ci700_tag_cache. metadata_controller30.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.metadata_controller31	ExportTest.CI700.ci700_tag_cachemetadata_controller31
CI700.ci700_tag_cache. metadata_controller31.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller32	ExportTest.CI700.ci700_tag_cachemetadata_controller32
CI700.ci700_tag_cache. metadata_controller32.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller33	ExportTest.CI700.ci700_tag_cachemetadata_controller33
CI700.ci700_tag_cache. metadata_controller33.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller34	ExportTest.CI700.ci700_tag_cachemetadata_controller34
CI700.ci700_tag_cache. metadata_controller34.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller35	ExportTest.CI700.ci700_tag_cachemetadata_controller35
CI700.ci700_tag_cache. metadata_controller35.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller36	ExportTest.CI700.ci700_tag_cachemetadata_controller36
CI700.ci700_tag_cache. metadata_controller36.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller37	ExportTest.CI700.ci700_tag_cachemetadata_controller37
CI700.ci700_tag_cache. metadata_controller37.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller38	ExportTest.CI700.ci700_tag_cachemetadata_controller38
CI700.ci700_tag_cache. metadata_controller38.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller39	ExportTest.CI700.ci700_tag_cachemetadata_controller39
CI700.ci700_tag_cache. metadata_controller39.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller4	ExportTest.CI700.ci700_tag_cachemetadata_controller4
CI700.ci700_tag_cache. metadata_controller4.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller40	ExportTest.CI700.ci700_tag_cachemetadata_controller40
CI700.ci700_tag_cache. metadata_controller40.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller41	ExportTest.CI700.ci700_tag_cachemetadata_controller41
CI700.ci700_tag_cache. metadata_controller41.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller42	ExportTest.CI700.ci700_tag_cachemetadata_controller42
CI700.ci700_tag_cache. metadata_controller42.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller43	ExportTest.CI700.ci700_tag_cachemetadata_controller43
CI700.ci700_tag_cache. metadata_controller43.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller44	ExportTest.CI700.ci700_tag_cachemetadata_controller44

InstanceName	ComponentName
CI700.ci700_tag_cache. metadata_controller44.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller45	ExportTest.CI700.ci700_tag_cachemetadata_controller45
CI700.ci700_tag_cache. metadata_controller45.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller46	ExportTest.CI700.ci700_tag_cachemetadata_controller46
CI700.ci700_tag_cache. metadata_controller46.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller47	ExportTest.CI700.ci700_tag_cachemetadata_controller47
CI700.ci700_tag_cache. metadata_controller47.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller48	ExportTest.CI700.ci700_tag_cachemetadata_controller48
CI700.ci700_tag_cache. metadata_controller48.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller49	ExportTest.CI700.ci700_tag_cachemetadata_controller49
CI700.ci700_tag_cache. metadata_controller49.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller5	ExportTest.CI700.ci700_tag_cachemetadata_controller5
CI700.ci700_tag_cache. metadata_controller5.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller50	ExportTest.CI700.ci700_tag_cachemetadata_controller50
CI700.ci700_tag_cache. metadata_controller50.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller51	ExportTest.CI700.ci700_tag_cachemetadata_controller51
CI700.ci700_tag_cache. metadata_controller51.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller52	ExportTest.CI700.ci700_tag_cachemetadata_controller52
CI700.ci700_tag_cache. metadata_controller52.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller53	ExportTest.CI700.ci700_tag_cachemetadata_controller53
CI700.ci700_tag_cache. metadata_controller53.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller54	ExportTest.CI700.ci700_tag_cachemetadata_controller54
CI700.ci700_tag_cache. metadata_controller54.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller55	ExportTest.CI700.ci700_tag_cachemetadata_controller55
CI700.ci700_tag_cache. metadata_controller55.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller56	ExportTest.CI700.ci700_tag_cachemetadata_controller56
CI700.ci700_tag_cache. metadata_controller56.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller57	ExportTest.CI700.ci700_tag_cachemetadata_controller57
CI700.ci700_tag_cache. metadata_controller57.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.metadata_controller58	ExportTest.CI700.ci700_tag_cachemetadata_controller58
CI700.ci700_tag_cache. metadata_controller58.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller59	ExportTest.CI700.ci700_tag_cachemetadata_controller59
CI700.ci700_tag_cache. metadata_controller59.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller6	ExportTest.CI700.ci700_tag_cachemetadata_controller6
CI700.ci700_tag_cache. metadata_controller6.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller60	ExportTest.CI700.ci700_tag_cachemetadata_controller60
CI700.ci700_tag_cache. metadata_controller60.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller61	ExportTest.CI700.ci700_tag_cachemetadata_controller61
CI700.ci700_tag_cache. metadata_controller61.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller62	ExportTest.CI700.ci700_tag_cachemetadata_controller62
CI700.ci700_tag_cache. metadata_controller62.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller63	ExportTest.CI700.ci700_tag_cachemetadata_controller63
CI700.ci700_tag_cache. metadata_controller63.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller64	ExportTest.CI700.ci700_tag_cachemetadata_controller64
CI700.ci700_tag_cache. metadata_controller64.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller65	ExportTest.CI700.ci700_tag_cachemetadata_controller65
CI700.ci700_tag_cache. metadata_controller65.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller66	ExportTest.CI700.ci700_tag_cachemetadata_controller66
CI700.ci700_tag_cache. metadata_controller66.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller67	ExportTest.CI700.ci700_tag_cachemetadata_controller67
CI700.ci700_tag_cache. metadata_controller67.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller68	ExportTest.CI700.ci700_tag_cachemetadata_controller68
CI700.ci700_tag_cache. metadata_controller68.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller69	ExportTest.CI700.ci700_tag_cachemetadata_controller69
CI700.ci700_tag_cache. metadata_controller69.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller7	ExportTest.CI700.ci700_tag_cachemetadata_controller7
CI700.ci700_tag_cache. metadata_controller7.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller70	ExportTest.CI700.ci700_tag_cachemetadata_controller70

InstanceName	ComponentName
CI700.ci700_tag_cache. metadata_controller70.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller71	ExportTest.CI700.ci700_tag_cachemetadata_controller71
CI700.ci700_tag_cache. metadata_controller71.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller72	ExportTest.CI700.ci700_tag_cachemetadata_controller72
CI700.ci700_tag_cache. metadata_controller72.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller73	ExportTest.CI700.ci700_tag_cachemetadata_controller73
CI700.ci700_tag_cache. metadata_controller73.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller74	ExportTest.CI700.ci700_tag_cachemetadata_controller74
CI700.ci700_tag_cache. metadata_controller74.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller75	ExportTest.CI700.ci700_tag_cachemetadata_controller75
CI700.ci700_tag_cache. metadata_controller75.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller76	ExportTest.CI700.ci700_tag_cachemetadata_controller76
CI700.ci700_tag_cache. metadata_controller76.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller77	ExportTest.CI700.ci700_tag_cachemetadata_controller77
CI700.ci700_tag_cache. metadata_controller77.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller78	ExportTest.CI700.ci700_tag_cachemetadata_controller78
CI700.ci700_tag_cache. metadata_controller78.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller79	ExportTest.CI700.ci700_tag_cachemetadata_controller79
CI700.ci700_tag_cache. metadata_controller79.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller8	ExportTest.CI700.ci700_tag_cachemetadata_controller8
CI700.ci700_tag_cache. metadata_controller8.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller80	ExportTest.CI700.ci700_tag_cachemetadata_controller80
CI700.ci700_tag_cache. metadata_controller80.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller81	ExportTest.CI700.ci700_tag_cachemetadata_controller81
CI700.ci700_tag_cache. metadata_controller81.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller82	ExportTest.CI700.ci700_tag_cachemetadata_controller82
CI700.ci700_tag_cache. metadata_controller82.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller83	ExportTest.CI700.ci700_tag_cachemetadata_controller83
CI700.ci700_tag_cache. metadata_controller83.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.metadata_controller84	ExportTest.CI700.ci700_tag_cachemetadata_controller84
CI700.ci700_tag_cache. metadata_controller84.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller85	ExportTest.CI700.ci700_tag_cachemetadata_controller85
CI700.ci700_tag_cache. metadata_controller85.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller86	ExportTest.CI700.ci700_tag_cachemetadata_controller86
CI700.ci700_tag_cache. metadata_controller86.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller87	ExportTest.CI700.ci700_tag_cachemetadata_controller87
CI700.ci700_tag_cache. metadata_controller87.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller88	ExportTest.CI700.ci700_tag_cachemetadata_controller88
CI700.ci700_tag_cache. metadata_controller88.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller89	ExportTest.CI700.ci700_tag_cachemetadata_controller89
CI700.ci700_tag_cache. metadata_controller89.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller9	ExportTest.CI700.ci700_tag_cachemetadata_controller9
CI700.ci700_tag_cache. metadata_controller9.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller90	ExportTest.CI700.ci700_tag_cachemetadata_controller90
CI700.ci700_tag_cache. metadata_controller90.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller91	ExportTest.CI700.ci700_tag_cachemetadata_controller91
CI700.ci700_tag_cache. metadata_controller91.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller92	ExportTest.CI700.ci700_tag_cachemetadata_controller92
CI700.ci700_tag_cache. metadata_controller92.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller93	ExportTest.CI700.ci700_tag_cachemetadata_controller93
CI700.ci700_tag_cache. metadata_controller93.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller94	ExportTest.CI700.ci700_tag_cachemetadata_controller94
CI700.ci700_tag_cache. metadata_controller94.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller95	ExportTest.CI700.ci700_tag_cachemetadata_controller95
CI700.ci700_tag_cache. metadata_controller95.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller96	ExportTest.CI700.ci700_tag_cachemetadata_controller96
CI700.ci700_tag_cache. metadata_controller96.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller97	ExportTest.CI700.ci700_tag_cachemetadata_controller97

InstanceName	ComponentName
CI700.ci700_tag_cache. metadata_controller97.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller98	ExportTest.CI700.ci700_tag_cachemetadata_controller98
CI700.ci700_tag_cache. metadata_controller98.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller99	ExportTest.CI700.ci700_tag_cachemetadata_controller99
CI700.ci700_tag_cache. metadata_controller99.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.remapper0	PVBusMapper
CI700.ci700_tag_cache.remapper1	PVBusMapper
CI700.ci700_tag_cache.remapper10	PVBusMapper
CI700.ci700_tag_cache.remapper100	PVBusMapper
CI700.ci700_tag_cache.remapper101	PVBusMapper
CI700.ci700_tag_cache.remapper102	PVBusMapper
CI700.ci700_tag_cache.remapper103	PVBusMapper
CI700.ci700_tag_cache.remapper104	PVBusMapper
CI700.ci700_tag_cache.remapper105	PVBusMapper
CI700.ci700_tag_cache.remapper106	PVBusMapper
CI700.ci700_tag_cache.remapper107	PVBusMapper
CI700.ci700_tag_cache.remapper108	PVBusMapper
CI700.ci700_tag_cache.remapper109	PVBusMapper
CI700.ci700_tag_cache.remapper11	PVBusMapper
CI700.ci700_tag_cache.remapper110	PVBusMapper
CI700.ci700_tag_cache.remapper111	PVBusMapper
CI700.ci700_tag_cache.remapper112	PVBusMapper
CI700.ci700_tag_cache.remapper113	PVBusMapper
CI700.ci700_tag_cache.remapper114	PVBusMapper
CI700.ci700_tag_cache.remapper115	PVBusMapper
CI700.ci700_tag_cache.remapper116	PVBusMapper
CI700.ci700_tag_cache.remapper117	PVBusMapper
CI700.ci700_tag_cache.remapper118	PVBusMapper
CI700.ci700_tag_cache.remapper119	PVBusMapper
CI700.ci700_tag_cache.remapper12	PVBusMapper
CI700.ci700_tag_cache.remapper120	PVBusMapper
CI700.ci700_tag_cache.remapper121	PVBusMapper
CI700.ci700_tag_cache.remapper122	PVBusMapper
CI700.ci700_tag_cache.remapper123	PVBusMapper
CI700.ci700_tag_cache.remapper124	PVBusMapper
CI700.ci700_tag_cache.remapper125	PVBusMapper
CI700.ci700_tag_cache.remapper126	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.remapper127	PVBusMapper
CI700.ci700_tag_cache.remapper13	PVBusMapper
CI700.ci700_tag_cache.remapper14	PVBusMapper
CI700.ci700_tag_cache.remapper15	PVBusMapper
CI700.ci700_tag_cache.remapper16	PVBusMapper
CI700.ci700_tag_cache.remapper17	PVBusMapper
CI700.ci700_tag_cache.remapper18	PVBusMapper
CI700.ci700_tag_cache.remapper19	PVBusMapper
CI700.ci700_tag_cache.remapper2	PVBusMapper
CI700.ci700_tag_cache.remapper20	PVBusMapper
CI700.ci700_tag_cache.remapper21	PVBusMapper
CI700.ci700_tag_cache.remapper22	PVBusMapper
CI700.ci700_tag_cache.remapper23	PVBusMapper
CI700.ci700_tag_cache.remapper24	PVBusMapper
CI700.ci700_tag_cache.remapper25	PVBusMapper
CI700.ci700_tag_cache.remapper26	PVBusMapper
CI700.ci700_tag_cache.remapper27	PVBusMapper
CI700.ci700_tag_cache.remapper28	PVBusMapper
CI700.ci700_tag_cache.remapper29	PVBusMapper
CI700.ci700_tag_cache.remapper3	PVBusMapper
CI700.ci700_tag_cache.remapper30	PVBusMapper
CI700.ci700_tag_cache.remapper31	PVBusMapper
CI700.ci700_tag_cache.remapper32	PVBusMapper
CI700.ci700_tag_cache.remapper33	PVBusMapper
CI700.ci700_tag_cache.remapper34	PVBusMapper
CI700.ci700_tag_cache.remapper35	PVBusMapper
CI700.ci700_tag_cache.remapper36	PVBusMapper
CI700.ci700_tag_cache.remapper37	PVBusMapper
CI700.ci700_tag_cache.remapper38	PVBusMapper
CI700.ci700_tag_cache.remapper39	PVBusMapper
CI700.ci700_tag_cache.remapper4	PVBusMapper
CI700.ci700_tag_cache.remapper40	PVBusMapper
CI700.ci700_tag_cache.remapper41	PVBusMapper
CI700.ci700_tag_cache.remapper42	PVBusMapper
CI700.ci700_tag_cache.remapper43	PVBusMapper
CI700.ci700_tag_cache.remapper44	PVBusMapper
CI700.ci700_tag_cache.remapper45	PVBusMapper
CI700.ci700_tag_cache.remapper46	PVBusMapper
CI700.ci700_tag_cache.remapper47	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.remapper48	PVBusMapper
CI700.ci700_tag_cache.remapper49	PVBusMapper
CI700.ci700_tag_cache.remapper5	PVBusMapper
CI700.ci700_tag_cache.remapper50	PVBusMapper
CI700.ci700_tag_cache.remapper51	PVBusMapper
CI700.ci700_tag_cache.remapper52	PVBusMapper
CI700.ci700_tag_cache.remapper53	PVBusMapper
CI700.ci700_tag_cache.remapper54	PVBusMapper
CI700.ci700_tag_cache.remapper55	PVBusMapper
CI700.ci700_tag_cache.remapper56	PVBusMapper
CI700.ci700_tag_cache.remapper57	PVBusMapper
CI700.ci700_tag_cache.remapper58	PVBusMapper
CI700.ci700_tag_cache.remapper59	PVBusMapper
CI700.ci700_tag_cache.remapper6	PVBusMapper
CI700.ci700_tag_cache.remapper60	PVBusMapper
CI700.ci700_tag_cache.remapper61	PVBusMapper
CI700.ci700_tag_cache.remapper62	PVBusMapper
CI700.ci700_tag_cache.remapper63	PVBusMapper
CI700.ci700_tag_cache.remapper64	PVBusMapper
CI700.ci700_tag_cache.remapper65	PVBusMapper
CI700.ci700_tag_cache.remapper66	PVBusMapper
CI700.ci700_tag_cache.remapper67	PVBusMapper
CI700.ci700_tag_cache.remapper68	PVBusMapper
CI700.ci700_tag_cache.remapper69	PVBusMapper
CI700.ci700_tag_cache.remapper7	PVBusMapper
CI700.ci700_tag_cache.remapper70	PVBusMapper
CI700.ci700_tag_cache.remapper71	PVBusMapper
CI700.ci700_tag_cache.remapper72	PVBusMapper
CI700.ci700_tag_cache.remapper73	PVBusMapper
CI700.ci700_tag_cache.remapper74	PVBusMapper
CI700.ci700_tag_cache.remapper75	PVBusMapper
CI700.ci700_tag_cache.remapper76	PVBusMapper
CI700.ci700_tag_cache.remapper77	PVBusMapper
CI700.ci700_tag_cache.remapper78	PVBusMapper
CI700.ci700_tag_cache.remapper79	PVBusMapper
CI700.ci700_tag_cache.remapper8	PVBusMapper
CI700.ci700_tag_cache.remapper80	PVBusMapper
CI700.ci700_tag_cache.remapper81	PVBusMapper
CI700.ci700_tag_cache.remapper82	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.remapper83	PVBusMapper
CI700.ci700_tag_cache.remapper84	PVBusMapper
CI700.ci700_tag_cache.remapper85	PVBusMapper
CI700.ci700_tag_cache.remapper86	PVBusMapper
CI700.ci700_tag_cache.remapper87	PVBusMapper
CI700.ci700_tag_cache.remapper88	PVBusMapper
CI700.ci700_tag_cache.remapper89	PVBusMapper
CI700.ci700_tag_cache.remapper9	PVBusMapper
CI700.ci700_tag_cache.remapper90	PVBusMapper
CI700.ci700_tag_cache.remapper91	PVBusMapper
CI700.ci700_tag_cache.remapper92	PVBusMapper
CI700.ci700_tag_cache.remapper93	PVBusMapper
CI700.ci700_tag_cache.remapper94	PVBusMapper
CI700.ci700_tag_cache.remapper95	PVBusMapper
CI700.ci700_tag_cache.remapper96	PVBusMapper
CI700.ci700_tag_cache.remapper97	PVBusMapper
CI700.ci700_tag_cache.remapper98	PVBusMapper
CI700.ci700_tag_cache.remapper99	PVBusMapper
CI700.cmn600_cache	PVCache
CI700.cmn600_cache.upstream[0]	PVBusSlave
CI700.cmn600_cache.upstream[1]	PVBusSlave
CI700.cmn600_cache.upstream[2]	PVBusSlave
CI700.cmn600_cache.upstream[3]	PVBusSlave
CI700.hnf_exclusive_monitor	PVBusExclusiveMonitor
CI700.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CI700.hni_exclusive_monitor0	PVBusExclusiveMonitor
CI700.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CI700.ocm_decoder	PVBusMapper
CI700.ocm_exclusive_monitor	PVBusExclusiveMonitor
CI700.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CI700.snf_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-870: CI700 MTI instances

InstanceName	ComponentName
CI700	CI700
CI700.bus_slave_ocm_NS	PVBusSlave
CI700.bus_slave_ocm_S	PVBusSlave
CI700.ci700_tag_cache	CMNTAGCACHECADI

InstanceName	ComponentName
CI700.ci700_tag_cache.metadata_controller0	ExportTest.CI700.ci700_tag_cachemetadata_controller0
CI700.ci700_tag_cache. metadata_controller0.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller1	ExportTest.CI700.ci700_tag_cachemetadata_controller1
CI700.ci700_tag_cache. metadata_controller1.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller10	ExportTest.CI700.ci700_tag_cachemetadata_controller10
CI700.ci700_tag_cache. metadata_controller10.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller100	ExportTest.CI700.ci700_tag_cachemetadata_controller100
CI700.ci700_tag_cache. metadata_controller100.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller101	ExportTest.CI700.ci700_tag_cachemetadata_controller101
CI700.ci700_tag_cache. metadata_controller101.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller102	ExportTest.CI700.ci700_tag_cachemetadata_controller102
CI700.ci700_tag_cache. metadata_controller102.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller103	ExportTest.CI700.ci700_tag_cachemetadata_controller103
CI700.ci700_tag_cache. metadata_controller103.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller104	ExportTest.CI700.ci700_tag_cachemetadata_controller104
CI700.ci700_tag_cache. metadata_controller104.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller105	ExportTest.CI700.ci700_tag_cachemetadata_controller105
CI700.ci700_tag_cache. metadata_controller105.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller106	ExportTest.CI700.ci700_tag_cachemetadata_controller106
CI700.ci700_tag_cache. metadata_controller106.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller107	ExportTest.CI700.ci700_tag_cachemetadata_controller107
CI700.ci700_tag_cache. metadata_controller107.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller108	ExportTest.CI700.ci700_tag_cachemetadata_controller108
CI700.ci700_tag_cache. metadata_controller108.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller109	ExportTest.CI700.ci700_tag_cachemetadata_controller109
CI700.ci700_tag_cache. metadata_controller109.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller11	ExportTest.CI700.ci700_tag_cachemetadata_controller11
CI700.ci700_tag_cache. metadata_controller11.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller110	ExportTest.CI700.ci700_tag_cachemetadata_controller110

InstanceName	ComponentName
CI700.ci700_tag_cache. metadata_controller110.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller111	ExportTest.CI700.ci700_tag_cachemetadata_controller111
CI700.ci700_tag_cache. metadata_controller111.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller112	ExportTest.CI700.ci700_tag_cachemetadata_controller112
CI700.ci700_tag_cache. metadata_controller112.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller113	ExportTest.CI700.ci700_tag_cachemetadata_controller113
CI700.ci700_tag_cache. metadata_controller113.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller114	ExportTest.CI700.ci700_tag_cachemetadata_controller114
CI700.ci700_tag_cache. metadata_controller114.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller115	ExportTest.CI700.ci700_tag_cachemetadata_controller115
CI700.ci700_tag_cache. metadata_controller115.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller116	ExportTest.CI700.ci700_tag_cachemetadata_controller116
CI700.ci700_tag_cache. metadata_controller116.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller117	ExportTest.CI700.ci700_tag_cachemetadata_controller117
CI700.ci700_tag_cache. metadata_controller117.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller118	ExportTest.CI700.ci700_tag_cachemetadata_controller118
CI700.ci700_tag_cache. metadata_controller118.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller119	ExportTest.CI700.ci700_tag_cachemetadata_controller119
CI700.ci700_tag_cache. metadata_controller119.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller12	ExportTest.CI700.ci700_tag_cachemetadata_controller12
CI700.ci700_tag_cache. metadata_controller12.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller120	ExportTest.CI700.ci700_tag_cachemetadata_controller120
CI700.ci700_tag_cache. metadata_controller120.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller121	ExportTest.CI700.ci700_tag_cachemetadata_controller121
CI700.ci700_tag_cache. metadata_controller121.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller122	ExportTest.CI700.ci700_tag_cachemetadata_controller122
CI700.ci700_tag_cache. metadata_controller122.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller123	ExportTest.CI700.ci700_tag_cachemetadata_controller123
CI700.ci700_tag_cache. metadata_controller123.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.metadata_controller124	ExportTest.CI700.ci700_tag_cachemetadata_controller124
CI700.ci700_tag_cache. metadata_controller124.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller125	ExportTest.CI700.ci700_tag_cachemetadata_controller125
CI700.ci700_tag_cache. metadata_controller125.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller126	ExportTest.CI700.ci700_tag_cachemetadata_controller126
CI700.ci700_tag_cache. metadata_controller126.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller127	ExportTest.CI700.ci700_tag_cachemetadata_controller127
CI700.ci700_tag_cache. metadata_controller127.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller13	ExportTest.CI700.ci700_tag_cachemetadata_controller13
CI700.ci700_tag_cache. metadata_controller13.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller14	ExportTest.CI700.ci700_tag_cachemetadata_controller14
CI700.ci700_tag_cache. metadata_controller14.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller15	ExportTest.CI700.ci700_tag_cachemetadata_controller15
CI700.ci700_tag_cache. metadata_controller15.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller16	ExportTest.CI700.ci700_tag_cachemetadata_controller16
CI700.ci700_tag_cache. metadata_controller16.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller17	ExportTest.CI700.ci700_tag_cachemetadata_controller17
CI700.ci700_tag_cache. metadata_controller17.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller18	ExportTest.CI700.ci700_tag_cachemetadata_controller18
CI700.ci700_tag_cache. metadata_controller18.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller19	ExportTest.CI700.ci700_tag_cachemetadata_controller19
CI700.ci700_tag_cache. metadata_controller19.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller2	ExportTest.CI700.ci700_tag_cachemetadata_controller2
CI700.ci700_tag_cache. metadata_controller2.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller20	ExportTest.CI700.ci700_tag_cachemetadata_controller20
CI700.ci700_tag_cache. metadata_controller20.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller21	ExportTest.CI700.ci700_tag_cachemetadata_controller21
CI700.ci700_tag_cache. metadata_controller21.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller22	ExportTest.CI700.ci700_tag_cachemetadata_controller22

InstanceName	ComponentName
CI700.ci700_tag_cache. metadata_controller22.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller23	ExportTest.CI700.ci700_tag_cachemetadata_controller23
CI700.ci700_tag_cache. metadata_controller23.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller24	ExportTest.CI700.ci700_tag_cachemetadata_controller24
CI700.ci700_tag_cache. metadata_controller24.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller25	ExportTest.CI700.ci700_tag_cachemetadata_controller25
CI700.ci700_tag_cache. metadata_controller25.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller26	ExportTest.CI700.ci700_tag_cachemetadata_controller26
CI700.ci700_tag_cache. metadata_controller26.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller27	ExportTest.CI700.ci700_tag_cachemetadata_controller27
CI700.ci700_tag_cache. metadata_controller27.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller28	ExportTest.CI700.ci700_tag_cachemetadata_controller28
CI700.ci700_tag_cache. metadata_controller28.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller29	ExportTest.CI700.ci700_tag_cachemetadata_controller29
CI700.ci700_tag_cache. metadata_controller29.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller3	ExportTest.CI700.ci700_tag_cachemetadata_controller3
CI700.ci700_tag_cache. metadata_controller3.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller30	ExportTest.CI700.ci700_tag_cachemetadata_controller30
CI700.ci700_tag_cache. metadata_controller30.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller31	ExportTest.CI700.ci700_tag_cachemetadata_controller31
CI700.ci700_tag_cache. metadata_controller31.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller32	ExportTest.CI700.ci700_tag_cachemetadata_controller32
CI700.ci700_tag_cache. metadata_controller32.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller33	ExportTest.CI700.ci700_tag_cachemetadata_controller33
CI700.ci700_tag_cache. metadata_controller33.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller34	ExportTest.CI700.ci700_tag_cachemetadata_controller34
CI700.ci700_tag_cache. metadata_controller34.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller35	ExportTest.CI700.ci700_tag_cachemetadata_controller35
CI700.ci700_tag_cache. metadata_controller35.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.metadata_controller36	ExportTest.CI700.ci700_tag_cachemetadata_controller36
CI700.ci700_tag_cache. metadata_controller36.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller37	ExportTest.CI700.ci700_tag_cachemetadata_controller37
CI700.ci700_tag_cache. metadata_controller37.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller38	ExportTest.CI700.ci700_tag_cachemetadata_controller38
CI700.ci700_tag_cache. metadata_controller38.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller39	ExportTest.CI700.ci700_tag_cachemetadata_controller39
CI700.ci700_tag_cache. metadata_controller39.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller4	ExportTest.CI700.ci700_tag_cachemetadata_controller4
CI700.ci700_tag_cache. metadata_controller4.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller40	ExportTest.CI700.ci700_tag_cachemetadata_controller40
CI700.ci700_tag_cache. metadata_controller40.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller41	ExportTest.CI700.ci700_tag_cachemetadata_controller41
CI700.ci700_tag_cache. metadata_controller41.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller42	ExportTest.CI700.ci700_tag_cachemetadata_controller42
CI700.ci700_tag_cache. metadata_controller42.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller43	ExportTest.CI700.ci700_tag_cachemetadata_controller43
CI700.ci700_tag_cache. metadata_controller43.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller44	ExportTest.CI700.ci700_tag_cachemetadata_controller44
CI700.ci700_tag_cache. metadata_controller44.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller45	ExportTest.CI700.ci700_tag_cachemetadata_controller45
CI700.ci700_tag_cache. metadata_controller45.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller46	ExportTest.CI700.ci700_tag_cachemetadata_controller46
CI700.ci700_tag_cache. metadata_controller46.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller47	ExportTest.CI700.ci700_tag_cachemetadata_controller47
CI700.ci700_tag_cache. metadata_controller47.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller48	ExportTest.CI700.ci700_tag_cachemetadata_controller48
CI700.ci700_tag_cache. metadata_controller48.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller49	ExportTest.CI700.ci700_tag_cachemetadata_controller49

InstanceName	ComponentName
CI700.ci700_tag_cache. metadata_controller49.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller5	ExportTest.CI700.ci700_tag_cachemetadata_controller5
CI700.ci700_tag_cache. metadata_controller5.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller50	ExportTest.CI700.ci700_tag_cachemetadata_controller50
CI700.ci700_tag_cache. metadata_controller50.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller51	ExportTest.CI700.ci700_tag_cachemetadata_controller51
CI700.ci700_tag_cache. metadata_controller51.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller52	ExportTest.CI700.ci700_tag_cachemetadata_controller52
CI700.ci700_tag_cache. metadata_controller52.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller53	ExportTest.CI700.ci700_tag_cachemetadata_controller53
CI700.ci700_tag_cache. metadata_controller53.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller54	ExportTest.CI700.ci700_tag_cachemetadata_controller54
CI700.ci700_tag_cache. metadata_controller54.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller55	ExportTest.CI700.ci700_tag_cachemetadata_controller55
CI700.ci700_tag_cache. metadata_controller55.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller56	ExportTest.CI700.ci700_tag_cachemetadata_controller56
CI700.ci700_tag_cache. metadata_controller56.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller57	ExportTest.CI700.ci700_tag_cachemetadata_controller57
CI700.ci700_tag_cache. metadata_controller57.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller58	ExportTest.CI700.ci700_tag_cachemetadata_controller58
CI700.ci700_tag_cache. metadata_controller58.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller59	ExportTest.CI700.ci700_tag_cachemetadata_controller59
CI700.ci700_tag_cache. metadata_controller59.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller6	ExportTest.CI700.ci700_tag_cachemetadata_controller6
CI700.ci700_tag_cache. metadata_controller6.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller60	ExportTest.CI700.ci700_tag_cachemetadata_controller60
CI700.ci700_tag_cache. metadata_controller60.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller61	ExportTest.CI700.ci700_tag_cachemetadata_controller61
CI700.ci700_tag_cache. metadata_controller61.MetadataMapper	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.metadata_controller62	ExportTest.CI700.ci700_tag_cachemetadata_controller62
CI700.ci700_tag_cache. metadata_controller62.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller63	ExportTest.CI700.ci700_tag_cachemetadata_controller63
CI700.ci700_tag_cache. metadata_controller63.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller64	ExportTest.CI700.ci700_tag_cachemetadata_controller64
CI700.ci700_tag_cache. metadata_controller64.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller65	ExportTest.CI700.ci700_tag_cachemetadata_controller65
CI700.ci700_tag_cache. metadata_controller65.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller66	ExportTest.CI700.ci700_tag_cachemetadata_controller66
CI700.ci700_tag_cache. metadata_controller66.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller67	ExportTest.CI700.ci700_tag_cachemetadata_controller67
CI700.ci700_tag_cache. metadata_controller67.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller68	ExportTest.CI700.ci700_tag_cachemetadata_controller68
CI700.ci700_tag_cache. metadata_controller68.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller69	ExportTest.CI700.ci700_tag_cachemetadata_controller69
CI700.ci700_tag_cache. metadata_controller69.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller7	ExportTest.CI700.ci700_tag_cachemetadata_controller7
CI700.ci700_tag_cache. metadata_controller7.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller70	ExportTest.CI700.ci700_tag_cachemetadata_controller70
CI700.ci700_tag_cache. metadata_controller70.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller71	ExportTest.CI700.ci700_tag_cachemetadata_controller71
CI700.ci700_tag_cache. metadata_controller71.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller72	ExportTest.CI700.ci700_tag_cachemetadata_controller72
CI700.ci700_tag_cache. metadata_controller72.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller73	ExportTest.CI700.ci700_tag_cachemetadata_controller73
CI700.ci700_tag_cache. metadata_controller73.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller74	ExportTest.CI700.ci700_tag_cachemetadata_controller74
CI700.ci700_tag_cache. metadata_controller74.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller75	ExportTest.CI700.ci700_tag_cachemetadata_controller75

InstanceName	ComponentName
CI700.ci700_tag_cache. metadata_controller75.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller76	ExportTest.CI700.ci700_tag_cachemetadata_controller76
CI700.ci700_tag_cache. metadata_controller76.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller77	ExportTest.CI700.ci700_tag_cachemetadata_controller77
CI700.ci700_tag_cache. metadata_controller77.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller78	ExportTest.CI700.ci700_tag_cachemetadata_controller78
CI700.ci700_tag_cache. metadata_controller78.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller79	ExportTest.CI700.ci700_tag_cachemetadata_controller79
CI700.ci700_tag_cache. metadata_controller79.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller8	ExportTest.CI700.ci700_tag_cachemetadata_controller8
CI700.ci700_tag_cache. metadata_controller8.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller80	ExportTest.CI700.ci700_tag_cachemetadata_controller80
CI700.ci700_tag_cache. metadata_controller80.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller81	ExportTest.CI700.ci700_tag_cachemetadata_controller81
CI700.ci700_tag_cache. metadata_controller81.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller82	ExportTest.CI700.ci700_tag_cachemetadata_controller82
CI700.ci700_tag_cache. metadata_controller82.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller83	ExportTest.CI700.ci700_tag_cachemetadata_controller83
CI700.ci700_tag_cache. metadata_controller83.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller84	ExportTest.CI700.ci700_tag_cachemetadata_controller84
CI700.ci700_tag_cache. metadata_controller84.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller85	ExportTest.CI700.ci700_tag_cachemetadata_controller85
CI700.ci700_tag_cache. metadata_controller85.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller86	ExportTest.CI700.ci700_tag_cachemetadata_controller86
CI700.ci700_tag_cache. metadata_controller86.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller87	ExportTest.CI700.ci700_tag_cachemetadata_controller87
CI700.ci700_tag_cache. metadata_controller87.MetaDataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller88	ExportTest.CI700.ci700_tag_cachemetadata_controller88
CI700.ci700_tag_cache. metadata_controller88.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.metadata_controller89	ExportTest.CI700.ci700_tag_cachemetadata_controller89
CI700.ci700_tag_cache. metadata_controller89.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller9	ExportTest.CI700.ci700_tag_cachemetadata_controller9
CI700.ci700_tag_cache. metadata_controller9.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller90	ExportTest.CI700.ci700_tag_cachemetadata_controller90
CI700.ci700_tag_cache. metadata_controller90.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller91	ExportTest.CI700.ci700_tag_cachemetadata_controller91
CI700.ci700_tag_cache. metadata_controller91.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller92	ExportTest.CI700.ci700_tag_cachemetadata_controller92
CI700.ci700_tag_cache. metadata_controller92.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller93	ExportTest.CI700.ci700_tag_cachemetadata_controller93
CI700.ci700_tag_cache. metadata_controller93.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller94	ExportTest.CI700.ci700_tag_cachemetadata_controller94
CI700.ci700_tag_cache. metadata_controller94.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller95	ExportTest.CI700.ci700_tag_cachemetadata_controller95
CI700.ci700_tag_cache. metadata_controller95.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller96	ExportTest.CI700.ci700_tag_cachemetadata_controller96
CI700.ci700_tag_cache. metadata_controller96.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller97	ExportTest.CI700.ci700_tag_cachemetadata_controller97
CI700.ci700_tag_cache. metadata_controller97.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller98	ExportTest.CI700.ci700_tag_cachemetadata_controller98
CI700.ci700_tag_cache. metadata_controller98.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.metadata_controller99	ExportTest.CI700.ci700_tag_cachemetadata_controller99
CI700.ci700_tag_cache. metadata_controller99.MetadataMapper	PVBusMapper
CI700.ci700_tag_cache.remapper0	PVBusMapper
CI700.ci700_tag_cache.remapper1	PVBusMapper
CI700.ci700_tag_cache.remapper10	PVBusMapper
CI700.ci700_tag_cache.remapper100	PVBusMapper
CI700.ci700_tag_cache.remapper101	PVBusMapper
CI700.ci700_tag_cache.remapper102	PVBusMapper
CI700.ci700_tag_cache.remapper103	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.remapper104	PVBusMapper
CI700.ci700_tag_cache.remapper105	PVBusMapper
CI700.ci700_tag_cache.remapper106	PVBusMapper
CI700.ci700_tag_cache.remapper107	PVBusMapper
CI700.ci700_tag_cache.remapper108	PVBusMapper
CI700.ci700_tag_cache.remapper109	PVBusMapper
CI700.ci700_tag_cache.remapper11	PVBusMapper
CI700.ci700_tag_cache.remapper110	PVBusMapper
CI700.ci700_tag_cache.remapper111	PVBusMapper
CI700.ci700_tag_cache.remapper112	PVBusMapper
CI700.ci700_tag_cache.remapper113	PVBusMapper
CI700.ci700_tag_cache.remapper114	PVBusMapper
CI700.ci700_tag_cache.remapper115	PVBusMapper
CI700.ci700_tag_cache.remapper116	PVBusMapper
CI700.ci700_tag_cache.remapper117	PVBusMapper
CI700.ci700_tag_cache.remapper118	PVBusMapper
CI700.ci700_tag_cache.remapper119	PVBusMapper
CI700.ci700_tag_cache.remapper12	PVBusMapper
CI700.ci700_tag_cache.remapper120	PVBusMapper
CI700.ci700_tag_cache.remapper121	PVBusMapper
CI700.ci700_tag_cache.remapper122	PVBusMapper
CI700.ci700_tag_cache.remapper123	PVBusMapper
CI700.ci700_tag_cache.remapper124	PVBusMapper
CI700.ci700_tag_cache.remapper125	PVBusMapper
CI700.ci700_tag_cache.remapper126	PVBusMapper
CI700.ci700_tag_cache.remapper127	PVBusMapper
CI700.ci700_tag_cache.remapper13	PVBusMapper
CI700.ci700_tag_cache.remapper14	PVBusMapper
CI700.ci700_tag_cache.remapper15	PVBusMapper
CI700.ci700_tag_cache.remapper16	PVBusMapper
CI700.ci700_tag_cache.remapper17	PVBusMapper
CI700.ci700_tag_cache.remapper18	PVBusMapper
CI700.ci700_tag_cache.remapper19	PVBusMapper
CI700.ci700_tag_cache.remapper2	PVBusMapper
CI700.ci700_tag_cache.remapper20	PVBusMapper
CI700.ci700_tag_cache.remapper21	PVBusMapper
CI700.ci700_tag_cache.remapper22	PVBusMapper
CI700.ci700_tag_cache.remapper23	PVBusMapper
CI700.ci700_tag_cache.remapper24	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.remapper25	PVBusMapper
CI700.ci700_tag_cache.remapper26	PVBusMapper
CI700.ci700_tag_cache.remapper27	PVBusMapper
CI700.ci700_tag_cache.remapper28	PVBusMapper
CI700.ci700_tag_cache.remapper29	PVBusMapper
CI700.ci700_tag_cache.remapper3	PVBusMapper
CI700.ci700_tag_cache.remapper30	PVBusMapper
CI700.ci700_tag_cache.remapper31	PVBusMapper
CI700.ci700_tag_cache.remapper32	PVBusMapper
CI700.ci700_tag_cache.remapper33	PVBusMapper
CI700.ci700_tag_cache.remapper34	PVBusMapper
CI700.ci700_tag_cache.remapper35	PVBusMapper
CI700.ci700_tag_cache.remapper36	PVBusMapper
CI700.ci700_tag_cache.remapper37	PVBusMapper
CI700.ci700_tag_cache.remapper38	PVBusMapper
CI700.ci700_tag_cache.remapper39	PVBusMapper
CI700.ci700_tag_cache.remapper4	PVBusMapper
CI700.ci700_tag_cache.remapper40	PVBusMapper
CI700.ci700_tag_cache.remapper41	PVBusMapper
CI700.ci700_tag_cache.remapper42	PVBusMapper
CI700.ci700_tag_cache.remapper43	PVBusMapper
CI700.ci700_tag_cache.remapper44	PVBusMapper
CI700.ci700_tag_cache.remapper45	PVBusMapper
CI700.ci700_tag_cache.remapper46	PVBusMapper
CI700.ci700_tag_cache.remapper47	PVBusMapper
CI700.ci700_tag_cache.remapper48	PVBusMapper
CI700.ci700_tag_cache.remapper49	PVBusMapper
CI700.ci700_tag_cache.remapper5	PVBusMapper
CI700.ci700_tag_cache.remapper50	PVBusMapper
CI700.ci700_tag_cache.remapper51	PVBusMapper
CI700.ci700_tag_cache.remapper52	PVBusMapper
CI700.ci700_tag_cache.remapper53	PVBusMapper
CI700.ci700_tag_cache.remapper54	PVBusMapper
CI700.ci700_tag_cache.remapper55	PVBusMapper
CI700.ci700_tag_cache.remapper56	PVBusMapper
CI700.ci700_tag_cache.remapper57	PVBusMapper
CI700.ci700_tag_cache.remapper58	PVBusMapper
CI700.ci700_tag_cache.remapper59	PVBusMapper
CI700.ci700_tag_cache.remapper6	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.remapper60	PVBusMapper
CI700.ci700_tag_cache.remapper61	PVBusMapper
CI700.ci700_tag_cache.remapper62	PVBusMapper
CI700.ci700_tag_cache.remapper63	PVBusMapper
CI700.ci700_tag_cache.remapper64	PVBusMapper
CI700.ci700_tag_cache.remapper65	PVBusMapper
CI700.ci700_tag_cache.remapper66	PVBusMapper
CI700.ci700_tag_cache.remapper67	PVBusMapper
CI700.ci700_tag_cache.remapper68	PVBusMapper
CI700.ci700_tag_cache.remapper69	PVBusMapper
CI700.ci700_tag_cache.remapper7	PVBusMapper
CI700.ci700_tag_cache.remapper70	PVBusMapper
CI700.ci700_tag_cache.remapper71	PVBusMapper
CI700.ci700_tag_cache.remapper72	PVBusMapper
CI700.ci700_tag_cache.remapper73	PVBusMapper
CI700.ci700_tag_cache.remapper74	PVBusMapper
CI700.ci700_tag_cache.remapper75	PVBusMapper
CI700.ci700_tag_cache.remapper76	PVBusMapper
CI700.ci700_tag_cache.remapper77	PVBusMapper
CI700.ci700_tag_cache.remapper78	PVBusMapper
CI700.ci700_tag_cache.remapper79	PVBusMapper
CI700.ci700_tag_cache.remapper8	PVBusMapper
CI700.ci700_tag_cache.remapper80	PVBusMapper
CI700.ci700_tag_cache.remapper81	PVBusMapper
CI700.ci700_tag_cache.remapper82	PVBusMapper
CI700.ci700_tag_cache.remapper83	PVBusMapper
CI700.ci700_tag_cache.remapper84	PVBusMapper
CI700.ci700_tag_cache.remapper85	PVBusMapper
CI700.ci700_tag_cache.remapper86	PVBusMapper
CI700.ci700_tag_cache.remapper87	PVBusMapper
CI700.ci700_tag_cache.remapper88	PVBusMapper
CI700.ci700_tag_cache.remapper89	PVBusMapper
CI700.ci700_tag_cache.remapper9	PVBusMapper
CI700.ci700_tag_cache.remapper90	PVBusMapper
CI700.ci700_tag_cache.remapper91	PVBusMapper
CI700.ci700_tag_cache.remapper92	PVBusMapper
CI700.ci700_tag_cache.remapper93	PVBusMapper
CI700.ci700_tag_cache.remapper94	PVBusMapper
CI700.ci700_tag_cache.remapper95	PVBusMapper

InstanceName	ComponentName
CI700.ci700_tag_cache.remapper96	PVBusMapper
CI700.ci700_tag_cache.remapper97	PVBusMapper
CI700.ci700_tag_cache.remapper98	PVBusMapper
CI700.ci700_tag_cache.remapper99	PVBusMapper
CI700.cmn600_cache	CMN600Cache
CI700.cmn600_cache.upstream[0]	PVBusSlave
CI700.cmn600_cache.upstream[1]	PVBusSlave
CI700.cmn600_cache.upstream[2]	PVBusSlave
CI700.cmn600_cache.upstream[3]	PVBusSlave
CI700.hnf_exclusive_monitor	PVBusExclusiveMonitor
CI700.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CI700.hni_exclusive_monitor0	PVBusExclusiveMonitor
CI700.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CI700.ocm_decoder	PVBusMapper
CI700.ocm_exclusive_monitor	PVBusExclusiveMonitor
CI700.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CI700.snf_mapper	PVBusMapper

CI700 contains the following CADI targets:

- CI700
- CMN_TAG_CACHE

Configuring and using the model

- Major IP revisions (rX) are modeled and are controlled by the `revision` parameter.
- Minor IP revisions (pY) are not modeled. There are no functional differences in the model between pY revisions, with the exception of the `periph_id2` register.
- To configure the model, you must have installed Arm® Socrates™. The `mesh_config_file` parameter defines the mesh placement of the CHI nodes. Set it to the name of the yml configuration file emitted by Socrates. You must use version r1p6-00rel0 or later of Socrates to generate the file. For more information about Socrates, see [Arm Socrates](#) on Arm Developer. To download it, contact [Arm Technical Support](#).
- Interconnect models are based on the TRM description and do not typically model RTL defects.
- The model revision is read from the topology specified by the `mesh_config_file` parameter.
- The mapping between the port number for rnf, rni/rnd, hni, and snf/sbsx interface ports and `nodeId` is based on the `nodeId` index. For example, RNF2 controls `pvbuss_s_rnf[2]`. Its index is specified in the `node_info` register as `logical_id`. Similar behavior can be expected for HN-I.

If both RN-D and RN-I nodes are present, then all starting rni ports are mapped to RN-D nodes and then the RN-I nodes. For example, for CI700 with two RN-D nodes, one RN-I node, and given each RN-I or RN-D node controls three interface ports, `pvbuss_s_rni[0-2]` maps to RND0, `pvbuss_s_rni[3-5]` maps to RND1 and `pvbuss_s_rni[6-8]` maps to RNI0.

Similarly, SN-F and SBSX nodes are mapped to `pvbbus_m_snf[]` ports where starting ports are mapped to SN-F and then SBSX nodes.

- Mapping `HN*_SLC_SIZE_PARAM` values to cache sizes:
 - 8: 0KB where `HN*_SLC_NUM_WAYS_PARAM=16`



This value is not supported by the model.

- 2: 128KB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - 1: 256KB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - 0: 512KB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - 1: 1MB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - 2: 2MB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - 3: 3MB where `HN*_SLC_NUM_WAYS_PARAM=12`
 - 3: 4MB where `HN*_SLC_NUM_WAYS_PARAM=16`
- The Address Based Flush (ABF) feature is available. After it has started, the ABF completes instantly. The ABF implementation has the following limitations:
 - In the Fast Model, any HN-F in the interconnect can perform an ABF on any address range even if that range is not part of the System Cache Group (SCG) that the HN-F belongs to.
 - Clearing the OCM with an ABF is not supported. A trace has been added to highlight this.
 - When `HNF_ABF_PR.abf_mode` is Reserved and `HNF_ABF_PR.abf_enable` is set, an error is printed and the ABF status register is set to complete successfully, but no operation takes place.
 - Because an ABF completes instantly, the `SF_PM_TRANSITION_ABORT` state cannot be reached. Contact [Arm Technical Support](#) with questions about support for this error condition.
 - The optional interrupt `INTREQPPU` is not supported.

Model limitations

- Out of scope:
 - PMU counters are not supported. Counter registers are implemented as RAZ.
 - QoS is not supported and all related registers are RAZ/WI.
 - Error injection and error generation are not supported. All error registers are RAZ/WI.
 - Power, clock, and interrupt signals are not supported.

The P-channel (power) signals `PREQ_LOGIC`, `PSTATE_LOGIC`, `PACCEPT_LOGIC`, `PDENY_LOGIC`, and `PACTIVE_LOGIC`, are not implemented. The model behaves as if `PSTATE_LOGIC` is 5'b00000. So the initial power state of the HN-F nodes is NOSFSLC/OFF. This is reflected in the reset values of `cmn_hns_ppu_pwpr`, which was named

`por_hnf_ppu_pwpr` in earlier IP revisions. Software must write the `*_ppu_pwpr` registers to change the power state. Both HAM and FAM power states enable all of the SLC ways.

- Protocol credits and flit buffers are not supported.
- Snoop filtering is not supported.
- Prefetch Target operations are not supported.
- The MTE feature has been minimally tested and the functionality cannot be guaranteed. MTE error injection and detection are not supported.
- An access to a reserved or nonexistent register does not abort. It returns `pv::Tx_Data::TX_OK` and is RAZ/WI.
- Topologies with three I/O masters connected through an RN-I bridge device have been tested. Configurations where multiple I/O masters share a single ACE-lite port have not been tested.
- RNSAMs external to the mesh network are not functionally supported.



Note

Model parameter `force_rnsam_internal` is true by default. If this parameter is set to false and the topology has an RNFx (non-ESAM), the external bit (bit[31]) of the RNSAM children pointer is enabled. This bit is the only functionality governed by this parameter.

- By default, HN-F hashing uses the address [MAX:12] instead of the actual address [MAX:6], due to the DMI mechanism in the model. Enable the parameter `enable_rnsam_to_hnf_wider_hash` to make hashing logic use the actual address [MAX:6], but this might reduce the simulation speed of the model.
- OKB SLC data RAM and tag RAM are not supported.
- Remote chip addresses incorrectly allocate a line in the SLC when `cache_state_modelled=true`. This can cause unexpected cache behavior that does not match the RTL, exhausting the SLC earlier or extra evictions.
- The fields in the `mxp_device_port_connect_info` register corresponding to the number of device credit slices and CAL credit slices are not supported.
- SN-F hashing in HNSAM is not supported. HNSAM SN-F hashing is a single memory range mapped to multiple SN-Fs. Example HNFSAM_Mem_Map:

```
0x0000001000 - 0x0000002000 SNF0 SNF1
```

The model supports mapping HN-F to a single SN-F. Example HNFSAM_Mem_Map:

```
0x0000001000 - 0x0000002000 SNF0
0x0000002000 - 0x0000003000 SNF1
```

- Only hashing granularities of 4096B and above are supported.
- HN-Fs with different SLC sizes in the same configuration are not supported.
- MPAM features are not supported in the model, and the configuration-dependent register reset values do not match the RTL. Some of the registers (`por_hnf_mpam_idr` to `por_hnf_mpam_mbwumon_idr`) shared between security modes are not present in MPAM_S.

- The MPAM reset values described in `por_hnf_mpam_ns_por_hnf_mpam_idr[31:24]` default to the not-supported state. The model parameter `hnf_mpam_idr_override` can be used to override this but does not implement any of the functionality.
- MPAM_S `secure_register_groups_override` is not supported.
- MTU `secure_register_groups_override` is not supported.
- Transactions to addresses unused by Device registers in the Configuration Register Space are always routed to the Configuration Space even when the SAM is configured to use those addresses.
- GIC communication over A4S ports is not supported.
- All RNs are assumed to have the same HN mapping. Programming RNSAMs differently between RNs is not supported.
- RNSAM cannot be programmed with the same HN-F target in a hashed and non-hashed memory region.
- Address bit masking is not supported in RNSAM or HN-F SAM.
- For an HTG/SCG, the model does not consider the secondary region if the primary region is not valid.
- HN-D is only permitted on device port P2 in a single-MXP configuration.
- Only the last RN-F in the mesh, which is the one with the highest logical id, can be controlled by `mxp_p[0-5]_syscoreq_ct1` registers. Also, incorrectly, it can be controlled from any XP.
- No support for RAS.
- HN-F SAM:
 - Address masking in default hash regions in HN-F SAM is not supported.
 - Hashing in default hash regions in HN-F SAM is not supported. However, trace `SNF_HASHING_Target_SNF` displays the target SNF as if 3SN/6SN hashing were enabled.



The transaction is always routed to the SN0 nodeid programmed in the SAM_CONTROL register.

Note

- Re-programming regions in HN-F SAM is not tested.
- Hashing across CCGs in HN-F SAM is not supported.

Ports for CI700

Table 3-871: Ports

Name	Protocol	Type	Description
<code>event_clusters</code>	Signal	Peer	CPU event communication signal from the clusters.
<code>pvbust_m_hni[4]</code>	PVBus	Master	HNI downstream ports.
<code>pvbust_m_snf[8]</code>	PVBus	Master	SNF downstream port.
<code>pvbust_s_apb</code>	PVBus	Slave	APB interface port.
<code>pvbust_s_rnf[8]</code>	PVBus	Slave	RNF upstream ports.

Name	Protocol	Type	Description
pvbus_s_rni[24]	PVBus	Slave	RNI upstream ports.
reset_in	Signal	Slave	Reset signal.
rnf_sci_s[8]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-F in and out of the coherency domain.
rnf_upstream_reset_in[8]	Signal	Slave	Used by the interconnect to determine the reset state of the RNF manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.
rni_sci_s[24]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-D in and out of the coherency domain.
rni_upstream_reset_in[24]	Signal	Slave	Used by the interconnect to determine the reset state of the RNI manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.

Parameters for CI700

acchannelen_rnf

DEPRECATED: Will be removed after FM 11.18. Use 'rnf_sci_enable' instead. For each rnf port, indicates if the port is populated with a snoop responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

acchannelen_rni

DEPRECATED: Will be removed after FM 11.18. Use 'rni_sci_enable' instead. For each rni port, indicates if the port is populated with a dvm responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

bypass_tag_cache

If true, CI700 will bypass the tag cache component which provides the MTE support

Type

bool

Default value

0x0

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x0

ci700_tag_cache.metadata_controller.init_value

Initialize metadata memory with this value. If one of `init_values_json` or `init_values_json_file` is specified, this value applies only to any metadata not specified in the JSON.

Type

int

Default value

0xd

ci700_tag_cache.metadata_controller.init_values_json

A JSON value describing initial metadata values. Mutually exclusive with `init_values_json_file`. The format is as follows: { "regions": [{ "begin": 0x0, "end": 0x10000, "mte_tag": 0xa }, { "begin": 0x20000, "end": 0x50000, "mte_tag": 0xc }] }

Type

string

Default value**ci700_tag_cache.metadata_controller.init_values_json_file**

Path to a JSON file with initial metadata values. Mutually exclusive with `init_values_json`. The format is as follows: { "regions": [{ "begin": 0x0, "end": 0x10000, "mte_tag": 0xa }, { "begin": 0x20000, "end": 0x50000, "mte_tag": 0xc }] }

Type

string

Default value**ci700_tag_cache.metadata_controller.is_enabled**

If false, disables the MetaData controller functionality, and makes the component invisible to passing transactions.

Type

bool

Default value

0x0

ci700_tag_cache.metadata_controller.mte_tag_carveout_json

JSON string that specifies the PA range of the tag carveout regions and the beginning of the PA range for which they provide tag storage. If `pa_regions_with_metadata_storage` defines which regions can have metadata, the tag carveout regions cannot overlap them, and each tagged region must be entirely covered by one of them. The block size must be

≥ 64 bytes and a power of 2, defaulting to 4KiB. The maximum block size supported is 4KiB. The carveout region size must be ≥ 4 KiB and a power of 2, and determines the size of the corresponding tagged region. { "regions": [{ "begin": 0x0, "tag_carveout_region": [0xfffff00000, 0xfffff00fff] }, { "begin": 0x20000, "tag_carveout_region": [0xfffff01000, 0xfffff01fff], "block_size": 0x100 }, { "begin": 0x100000, "tag_carveout_region": [0xfffff08000, 0xfffff0Bfff], "block_size": 0x2000 },] }

Type

string

Default value**ci700_tag_cache.metadata_controller.mte_tag_carveout_json_file**

Path to a JSON file that specifies the PA range of the tag carveout regions with the same format as mte_tag_carveout_json. Only one of mte_tag_carveout_json and mte_tag_carveout_json_file can be used.

Type

string

Default value**ci700_tag_cache.metadata_controller.mte_tag_carveout_tag_order**

Order of the tags within the MTE tag carveout blocks. This can be little-endian (same order as the corresponding tagged data) or big-endian (reverse order). The parameter accepts both '-' and '_', so 'little-endian', 'big-endian', 'little_endian' and 'big_endian' are all valid. THIS PARAMETER HAS NO FUNCTIONALITY AT THE MOMENT.

Type

string

Default value

little-endian

ci700_tag_cache.metadata_controller.pa_regions_with_metadata_storage

Specify the address region where the metadata storage is available for each PAS in a JSON format. If the PAS does not have a region specified, the PAS has metadata storage for all of the space. The regions are defined by begin and end_incl addresses. Example: { "ns": [0xa0000000, 0xa0000fff], "s": [0xb0000000, 0xb0000fff], "rl": [0xc0000000, 0xc0000fff], "rt": [0xd0000000, 0xd0000fff] } ns: non-secure, s: secure, rl: realm, rt: root

Type

string

Default value**debug_force_snoop**

The CI700 interconnect will normally start with snooping disabled. The parameter rnf_sci_enable and rni_sci_enable determines which connections are managed by the System Coherency Interface. For connections that are managed by the System Coherency Interface, then they will look after themselves for entering and leaving the coherency domain and this

parameter has no effect for those ports. However, for connections that are `_not_` managed by the System Coherency Interface, this parameter enables the upstream system of a port to be snooped if the upstream is not in reset and if `acchannelen_rnf` allows it. NOTE: This parameter is for debug purpose only Do not use this parameter instead of correctly configuring CMN. Refer CI700 documentation page for more info about this parameter in fast models reference manual

Type

bool

Default value

0x0

`enable_logger`

Enable PVBUSLoggers for the downstream ports in the CMN model.

Type

bool

Default value

0x0

`enable_rnsam_to_hnf_wider_hash`

Enable support of wider hash for the RNSAM to HNF communication. If this variable enabled, then bits[47:6] from PA used in hashing function. By default it is [47:12].

Type

bool

Default value

0x0

`force_rnsam_internal`

Force all RNSAMs to be internal independently of the mesh topology.

Type

bool

Default value

0x1

`hnf_mpam_idr_override`

Set to override `hnf_mpam_idr[31:24]` value. Bit[28] is Reserved and is ignored.

Type

int

Default value

0x0

`mesh_config_file`

Name of a file containing mesh placement of CI700 components.

Type

string

Default value

/tmp/plgbuild/abs_build/1378878_61223/trunk/work/fastsim/autodocs/../../Platforms/
LISA/FVP_Base/Build_AEMvA-AEMvA-CI700/ci700.yml

periphbase

Value for PERIPHBASE. Bits [27:0] are treated 0

Type

int

Default value

0x20000000

print_cmnn_ccix_config

Print information about the CCIX configuration.

Type

bool

Default value

0x0

print_cmnn_config

Print the mesh topology and children pointers acquired from the YAML file.

Type

bool

Default value

0x0

revision

Component revision. Currently supports r2p0, r1p0, r0p0. The version in the mesh_config_file takes priority.

Type

string

Default value

r0p0

rnf_sci_enable

For each rnf port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

rni_sci_enable

For each rni port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

show_banner

Show component banner: 0=>supress entire banner, 1=>suppress config file, 2+=>show full banner.

Type

int

Default value

0x2

skip_cmnn_config_check

Skip any topology configuration checks. The maximum number of devices per type not verified.

Type

bool

Default value

0x0

use_yaml_periphbase

Use yaml param CFGM_PERIPHBASE_PARAM to specify periphbase address. If false, model parameter 'periphbase' will be used.

Type

bool

Default value

0x0

yaml_has_node_addresses

Does the top-level YML file describe node-addresses ?

Type

bool

Default value

0x0

3.10.13 CMN600

CMN600 Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-872: IP revisions support

Revision	Quality level
r1p1	Full support
r3p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CMN600

This model has the following Iris instances:

Table 3-873: CMN600 Iris instances

InstanceName	ComponentName
CMN600	CMN600
CMN600.bus_slave_ocm_NS	PVBusSlave
CMN600.bus_slave_ocm_S	PVBusSlave
CMN600.cmn600_cache	CMN600Cache
CMN600.cmn600_cache.upstream[0]	PVBusSlave
CMN600.cmn600_cache.upstream[10]	PVBusSlave
CMN600.cmn600_cache.upstream[11]	PVBusSlave
CMN600.cmn600_cache.upstream[12]	PVBusSlave
CMN600.cmn600_cache.upstream[13]	PVBusSlave
CMN600.cmn600_cache.upstream[14]	PVBusSlave
CMN600.cmn600_cache.upstream[15]	PVBusSlave
CMN600.cmn600_cache.upstream[16]	PVBusSlave
CMN600.cmn600_cache.upstream[17]	PVBusSlave
CMN600.cmn600_cache.upstream[1]	PVBusSlave
CMN600.cmn600_cache.upstream[2]	PVBusSlave
CMN600.cmn600_cache.upstream[3]	PVBusSlave
CMN600.cmn600_cache.upstream[4]	PVBusSlave
CMN600.cmn600_cache.upstream[5]	PVBusSlave
CMN600.cmn600_cache.upstream[6]	PVBusSlave
CMN600.cmn600_cache.upstream[7]	PVBusSlave
CMN600.cmn600_cache.upstream[8]	PVBusSlave
CMN600.cmn600_cache.upstream[9]	PVBusSlave
CMN600.hnf_exclusive_monitor	PVBusExclusiveMonitor

InstanceName	ComponentName
CMN600.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CMN600.hni_exclusive_monitor0	PVBusExclusiveMonitor
CMN600.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CMN600.ocm_decoder	PVBusMapper
CMN600.ocm_exclusive_monitor	PVBusExclusiveMonitor
CMN600.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CMN600.snf_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-874: CMN600 MTI instances

InstanceName	ComponentName
CMN600	CMN600
CMN600.bus_slave_ocm_NS	PVBusSlave
CMN600.bus_slave_ocm_S	PVBusSlave
CMN600.cmn600_cache	CMN600Cache
CMN600.cmn600_cache.upstream[0]	PVBusSlave
CMN600.cmn600_cache.upstream[10]	PVBusSlave
CMN600.cmn600_cache.upstream[11]	PVBusSlave
CMN600.cmn600_cache.upstream[12]	PVBusSlave
CMN600.cmn600_cache.upstream[13]	PVBusSlave
CMN600.cmn600_cache.upstream[14]	PVBusSlave
CMN600.cmn600_cache.upstream[15]	PVBusSlave
CMN600.cmn600_cache.upstream[16]	PVBusSlave
CMN600.cmn600_cache.upstream[17]	PVBusSlave
CMN600.cmn600_cache.upstream[1]	PVBusSlave
CMN600.cmn600_cache.upstream[2]	PVBusSlave
CMN600.cmn600_cache.upstream[3]	PVBusSlave
CMN600.cmn600_cache.upstream[4]	PVBusSlave
CMN600.cmn600_cache.upstream[5]	PVBusSlave
CMN600.cmn600_cache.upstream[6]	PVBusSlave
CMN600.cmn600_cache.upstream[7]	PVBusSlave
CMN600.cmn600_cache.upstream[8]	PVBusSlave
CMN600.cmn600_cache.upstream[9]	PVBusSlave
CMN600.hnf_exclusive_monitor	PVBusExclusiveMonitor
CMN600.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CMN600.hni_exclusive_monitor0	PVBusExclusiveMonitor
CMN600.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CMN600.ocm_decoder	PVBusMapper
CMN600.ocm_exclusive_monitor	PVBusExclusiveMonitor

InstanceName	ComponentName
CMN600.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CMN600.snf_mapper	PVBusMapper

CMN600 contains the following CADI targets:

- CMN600
- CMN600Cache

About CMN600

- For information about the changes in this release, see the Fast Models Portfolio release notes.
- Major IP revisions (rX) are modeled and are controlled by the `revision` parameter.
- Minor IP revisions (pY) are not modeled. There are no functional differences in the model between pY revisions, with the exception of the `periph_id2` register.
- To configure the model, you must have installed Arm® Socrates™. The `mesh_config_file` parameter defines the mesh placement of the CHI nodes. Set it to the name of the yml configuration file emitted by Socrates. You must use version r1p4-01rel0 or later of Socrates to generate the file. For more information about Socrates, see [Arm Socrates](#) on Arm Developer. To download it, contact [Arm Technical Support](#).
- Interconnect models are based on the TRM description and do not typically model RTL defects.

The model supports the following features:

- rnf, rni/rnd, hni, and snf/sbsx interface ports. The mapping between the port number and `nodeId` is based on the `nodeId` index. For example, RNF2 controls `pvbuss_s_rnf[2]`. Its index is specified in the `node_info` register as `logical_id`. Similar behavior can be expected for HN-I.

If both RN-D and RN-I nodes are present, then all starting rni ports are mapped to RN-D nodes and then the RN-I nodes. For example, for CMN600 with two RN-D nodes, one RN-I node, and given each RN-I or RN-D node controls three interface ports, `pvbuss_s_rni[0-2]` maps to RND0, `pvbuss_s_rni[3-5]` maps to RND1 and `pvbuss_s_rni[6-8]` maps to RNI0.

Similarly, SN-F and SBSX nodes are mapped to `pvbuss_m_snf[]` ports where starting ports are mapped to SN-F and then SBSX nodes.

- Mapping `HN*_SLC_SIZE_PARAM` values to cache sizes:
 - -8: OKB where `HN*_SLC_NUM_WAYS_PARAM=16`



This value is not supported by the model.

- -2: 128KB where `HN*_SLC_NUM_WAYS_PARAM=16`
- -1: 256KB where `HN*_SLC_NUM_WAYS_PARAM=16`
- 0: 512KB where `HN*_SLC_NUM_WAYS_PARAM=16`

- 1: 1MB where `HN*_SLC_NUM_WAYS_PARAM=16`
- 2: 2MB where `HN*_SLC_NUM_WAYS_PARAM=16`
- 3: 3MB where `HN*_SLC_NUM_WAYS_PARAM=12`
- 3: 4MB where `HN*_SLC_NUM_WAYS_PARAM=16`
- The Address Based Flush (ABF) feature is available. After it has started, the ABF completes instantly. The ABF implementation has the following limitations:
 - In the Fast Model, any HN-F in the interconnect can perform an ABF on any address range even if that range is not part of the System Cache Group (SCG) that the HN-F belongs to.
 - Clearing the OCM with an ABF is not supported. A trace has been added to highlight this.
 - When `HN*_ABF_PR.abf_mode` is Reserved and `HN*_ABF_PR.abf_enable` is set, an error is printed and the ABF status register is set to complete successfully, but no operation takes place.
 - Because an ABF completes instantly, the `SF_PM_TRANSITION_ABORT` state cannot be reached. Contact [Arm Technical Support](#) with questions about support for this error condition.
 - The optional interrupt `INTREQPPU` is not supported.

Model limitations

- Out of scope:
 - PMU counters are not supported. Counter registers are implemented as RAZ.
 - QoS is not supported and all related registers are RAZ/WI.
 - Error injection and error generation are not supported. All error registers are RAZ/WI.
 - Power, clock, and interrupt signals are not supported.

The P-channel (power) signals `PREQ_LOGIC`, `PSTATE_LOGIC`, `PACCEPT_LOGIC`, `PDENY_LOGIC`, and `PACTIVE_LOGIC`, are not implemented. The model behaves as if `PSTATE_LOGIC` is 5'b00000. So the initial power state of the HN-F nodes is `NOSFSLC/OFF`. This is reflected in the reset values of `cmn_hns_ppu_pwpr`, which was named `por_hnf_ppu_pwpr` in earlier IP revisions. Software must write the `*_ppu_pwpr` registers to change the power state. Both HAM and FAM power states enable all of the SLC ways.

- Protocol credits and flit buffers are not supported.
- Snoop filtering is not supported.
- Prefetch Target operations are not supported.
- The MTE feature has been minimally tested and the functionality cannot be guaranteed. MTE error injection and detection are not supported.
- An access to a reserved or nonexistent register does not abort. It returns `pv::Tx_Data::TX_OK` and is RAZ/WI.
- Topologies with three I/O masters connected through an RN-I bridge device have been tested. Configurations where multiple I/O masters share a single ACE-lite port have not been tested.
- RNSAMs external to the mesh network are not functionally supported.



Note

Model parameter `force_rnsam_internal` is true by default. If this parameter is set to false and the topology has an RNFx (non-ESAM), the external bit (bit[31]) of the RNSAM children pointer is enabled. This bit is the only functionality governed by this parameter.

- By default, HN-F hashing uses the address [MAX:12] instead of the actual address [MAX:6], due to the DMI mechanism in the model. Enable the parameter `enable_rnsam_to_hnf_wider_hash` to make hashing logic use the actual address [MAX:6], but this might reduce the simulation speed of the model.
- OKB SLC data RAM and tag RAM are not supported.
- Source-based SLC cache partitioning is not supported.
- Way-based SLC cache partitioning is not supported.
- SLC/LCC partitioning dynamic allocation is not supported.
- Remote chip addresses incorrectly allocate a line in the SLC when `cache_state_modelled=true`. This can cause unexpected cache behavior that does not match the RTL, exhausting the SLC earlier or extra evictions.
- The fields in the `mvp_device_port_connect_info` register corresponding to the number of device credit slices and the number of CAL credit slices are not supported.
- SN-F hashing in HNSAM is not supported. HNSAM SN-F hashing is a single memory range mapped to multiple SN-Fs. Example HNFSAM_Mem_Map:

```
0x00000001000 - 0x00000002000 SNF0 SNF1
```

The model supports mapping HN-F to a single SN-F. Example HNFSAM_Mem_Map:

```
0x00000001000 - 0x00000002000 SNF0
0x00000002000 - 0x00000003000 SNF1
```

- HN-Fs with different SLC sizes in the same configuration are not supported.
- Transactions to addresses unused by Device registers in the Configuration Register Space are always routed to the Configuration Space even when the SAM is configured to use those addresses.
- GIC communication over A4S ports is not supported.
- CAL (Component Aggregation Layer) r2 and r3 features are supported. These features have limited testing.
- No specific Debug Trace (DT) node functionality is supported. See [4. Plug-ins for Fast Models](#) on page 4138 for information about using plug-ins for trace.
- For Power State Transition related flush/disable of the SLC due to HNF_PPU_PWPR register, only the Operational Mode field (bits[7:4]) is considered. The Power Policy field (bits[3:0]) is ignored.

- Setting HAM mode for the Operational Mode field (bits[7:4]) of the HNF_PPU_PWPR is equivalent to FAM mode. The SLCH2 is neither flushed nor disabled according to the Power Policy field (bits[3:0]).
- Only hashing granularities of 4096B and above are supported.
- All RNs are assumed to have the same HN mapping. Programming RNSAMs differently between RNs is not supported.
- RNSAM cannot be programmed with the same HN-F target in a hashed and non-hashed memory region.
- Address bit masking is not supported in RNSAM or HN-F SAM.
- The OCM does not keep track of the total memory used across the S/NS security world and does not perform any eviction either. It is up to the software to make sure it does not oversubscribe the OCM.
- In revision r3p0, for an HTG/SCG, the model does not consider the secondary region if the primary region is not valid.
- There is no support for RAS.
- The following limitations apply to System Cache Groups and Hash Target Groups:
 - A mix of local and remote HN-F targets is not supported.
 - CCG/CXRH target IDs in the HN target ID table are not supported (`sys_cache_grp_hn_nodeid_reg`).
 - The `hashed_target_grp_hnp_nodeid_reg` which is used for CCG/HN-P target IDs is not supported.
 - System/Hash Target Groups only support HN-Fs.
 - AXID hashing across HN-P/CCGs is not supported.
- HN-F SAM:
 - Address masking in default hash regions in HN-F SAM is not supported.
 - Hashing in default hash regions in HN-F SAM is not supported. However, trace `SNF_HASHING_Target_SNF` displays the target SNF as if 3SN/6SN hashing were enabled.



The transaction is always routed to the SNO nodeid programmed in the SAM_CONTROL register.

-
- Re-programming regions in HN-F SAM is not tested.
 - Hashing across CCGs in HN-F SAM is not supported.

About the `debug_force_snoop` parameter

The CMN interconnect normally starts up with snooping disabled.

The parameters `rnf_sci_enable` and `rni_sci_enable` determine which connections are managed by the System Coherency Interface. Connections that are managed by the System

Coherency Interface look after themselves for entering and leaving the coherency domain and `debug_force_snoop` has no effect on those ports.

However, for connections that are not managed by the System Coherency Interface, `debug_force_snoop` enables the system upstream of a port to be snooped if the upstream is not in reset and if `acchannelen_rnf` allows it.

Therefore, software does not need to know how to turn snoops in the interconnect on and off. The reset state of the upstream rnf ports is determined by the `rnf_upstream_reset_state[]` signals and must be connected for any port that could go into reset and you want managed by `debug_force_snoop`.

If the connection is managed by SCI or can never go into reset then you need not connect this and the interconnect assumes that the upstream system is not in reset and will always enable the snoops if `ACCHANNELENSx` allows it.

If you fail to connect `rnf_upstream_reset_state[]` correctly, then the upstream system might receive snoop messages while in reset and it will complain that it received a snoop request while it was in reset.

If software overrides the enables then the next change to `rnf_upstream_reset_state[]` or the interconnect reset will overwrite the software values.

Using this operation along with software controlling the enables could confuse the software. This option does not remove the responsibility of the upstream system to clean any shared dirty data from its caches before going into reset.



This parameter is for debug purposes only. Do not use it instead of correctly configuring CMN.

Ports for CMN600

Table 3-875: Ports

Name	Protocol	Type	Description
<code>event_clusters</code>	Signal	Peer	CPU event communication signal from the clusters.
<code>event_downstream_link_signal[6]</code>	Signal	Master	CPU event communication signal to the CMLHub. Event from the CMN towards the Hub NOTE that these are virtual "links" on underlying PCIe hardware bus.
<code>event_upstream_link_signal[6]</code>	Signal	Slave	Event from the Hub towards the CMN
<code>pvbust_m_cml</code>	PVBus	Master	CML downstream ports
<code>pvbust_m_cml_cfg</code>	PVBus	Master	CML downstream hub configuration port
<code>pvbust_m_hni[8]</code>	PVBus	Master	HNI downstream ports.
<code>pvbust_m_snf[16]</code>	PVBus	Master	SNF downstream port.
<code>pvbust_s_cml</code>	PVBus	Slave	CML upstream ports

Name	Protocol	Type	Description
pvbus_s_rnf[64]	PVBus	Slave	RNF upstream ports.
pvbus_s_rni[96]	PVBus	Slave	RNI upstream ports.
reset_in	Signal	Slave	Reset signal.
rnf_sci_s[64]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-F in and out of the coherency domain.
rnf_upstream_reset_in[64]	Signal	Slave	Used by the interconnect to determine the reset state of the RNF manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.
rni_sci_s[96]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-D in and out of the coherency domain.
rni_upstream_reset_in[96]	Signal	Slave	Used by the interconnect to determine the reset state of the RNI manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.

Parameters for CMN600

acchannelen_rnf

DEPRECATED: Will be removed after FM 11.18. Use 'rnf_sci_enable' instead. For each rnf port, indicates if the port is populated with a snoop responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

acchannelen_rni

DEPRECATED: Will be removed after FM 11.18. Use 'rni_sci_enable' instead. For each rni port, indicates if the port is populated with a dvm responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x0

debug_force_snoop

The CMN600 interconnect will normally start with snooping disabled. The parameter `rnf_sci_enable` and `rni_sci_enable` determines which connections are managed by the System Coherency Interface. For connections that are managed by the System Coherency Interface, then they will look after themselves for entering and leaving the coherency domain and this parameter has no effect for those ports. However, for connections that are `_not_` managed by the System Coherency Interface, this parameter enables the upstream system of a port to be snooped if the upstream is not in reset and if `acchannelen_rnf` allows it. NOTE: This parameter is for debug purpose only Do not use this parameter instead of correctly configuring CMN. Refer CMN600 documentation page for more info about this parameter in fast models reference manual

Type

bool

Default value

0x0

disable_CML_port

If true the model won't connect the CML port when there are nodes which support the CML feature (ie CXG nodes).

Type

bool

Default value

0x0

enable_logger

Enable PVBUSLoggers for the downstream ports in the CMN model.

Type

bool

Default value

0x0

enable_rnsam_to_hnf_wider_hash

Enable support of wider hash for the RNSAM to HNF communication. If this variable enabled, then bits[47:6] from PA used in hashing function. By default it is [47:12].

Type

bool

Default value

0x0

force_rnsam_internal

Force all RNSAMs to be internal independently of the mesh topology.

Type

bool

Default value

0x1

mesh_config_file

Name of a file containing mesh placement of CMN600 components.

Type

string

Default value/tmp/plgbuild/abs_build/1378878_61223/trunk/work/fastsim/autodocs/../../Platforms/
LISA/CSS/Kits4/RD_N1_E1_cmn600.yml**periphbase**

Value for PERIPHBASE. Bits [25:0] are treated 0

Type

int

Default value

0x20000000

print_cmn_ccix_config

Print information about the CCIX configuration.

Type

bool

Default value

0x0

print_cmn_config

Print the mesh topology and children pointers acquired from the YAML file.

Type

bool

Default value

0x0

revision

Component revision. Currently supports r1p1, r3p0.

Type

string

Default value

r1p1

rnf_sci_enable

For each rnf port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

rni_sci_enable

For each rni port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

show_banner

Show component banner: 0=>supress entire banner, 1=>suppress config file, 2+=>show full banner.

Type

int

Default value

0x2

skip_cmnn_config_check

Skip any topology configuration checks. The maximum number of devices per type not verified.

Type

bool

Default value

0x0

use_yaml_periphbase

Use yaml param CFGM_PERIPHBASE_PARAM to specify periphbase address. If false, model parameter 'periphbase' will be used.

Type

bool

Default value

0x0

3.10.14 CMN600AE

CMN600AE Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-876: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CMN600AE

This model has the following Iris instances:

Table 3-877: CMN600AE Iris instances

InstanceName	ComponentName
CMN600AE	CMN600AE
CMN600AE.bus_slave_ocm_NS	PVBusSlave
CMN600AE.bus_slave_ocm_S	PVBusSlave
CMN600AE.cmn600_cache	PVCache
CMN600AE.cmn600_cache.upstream[0]	PVBusSlave
CMN600AE.cmn600_cache.upstream[10]	PVBusSlave
CMN600AE.cmn600_cache.upstream[11]	PVBusSlave
CMN600AE.cmn600_cache.upstream[12]	PVBusSlave
CMN600AE.cmn600_cache.upstream[13]	PVBusSlave
CMN600AE.cmn600_cache.upstream[14]	PVBusSlave
CMN600AE.cmn600_cache.upstream[15]	PVBusSlave
CMN600AE.cmn600_cache.upstream[16]	PVBusSlave
CMN600AE.cmn600_cache.upstream[17]	PVBusSlave
CMN600AE.cmn600_cache.upstream[18]	PVBusSlave
CMN600AE.cmn600_cache.upstream[1]	PVBusSlave
CMN600AE.cmn600_cache.upstream[2]	PVBusSlave
CMN600AE.cmn600_cache.upstream[3]	PVBusSlave
CMN600AE.cmn600_cache.upstream[4]	PVBusSlave
CMN600AE.cmn600_cache.upstream[5]	PVBusSlave
CMN600AE.cmn600_cache.upstream[6]	PVBusSlave
CMN600AE.cmn600_cache.upstream[7]	PVBusSlave
CMN600AE.cmn600_cache.upstream[8]	PVBusSlave
CMN600AE.cmn600_cache.upstream[9]	PVBusSlave
CMN600AE.hnf_exclusive_monitor	PVBusExclusiveMonitor

InstanceName	ComponentName
CMN600AE.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CMN600AE.hni_exclusive_monitor0	PVBusExclusiveMonitor
CMN600AE.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CMN600AE.hni_exclusive_monitor1	PVBusExclusiveMonitor
CMN600AE.hni_exclusive_monitor1.bus_mapper	PVBusMapper
CMN600AE.mpu	PVBusMapper
CMN600AE.ocm_decoder	PVBusMapper
CMN600AE.ocm_exclusive_monitor	PVBusExclusiveMonitor
CMN600AE.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CMN600AE.snf_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-878: CMN600AE MTI instances

InstanceName	ComponentName
CMN600AE	CMN600AE
CMN600AE.bus_slave_ocm_NS	PVBusSlave
CMN600AE.bus_slave_ocm_S	PVBusSlave
CMN600AE.cmn600_cache	CMN600Cache
CMN600AE.cmn600_cache.upstream[0]	PVBusSlave
CMN600AE.cmn600_cache.upstream[10]	PVBusSlave
CMN600AE.cmn600_cache.upstream[11]	PVBusSlave
CMN600AE.cmn600_cache.upstream[12]	PVBusSlave
CMN600AE.cmn600_cache.upstream[13]	PVBusSlave
CMN600AE.cmn600_cache.upstream[14]	PVBusSlave
CMN600AE.cmn600_cache.upstream[15]	PVBusSlave
CMN600AE.cmn600_cache.upstream[16]	PVBusSlave
CMN600AE.cmn600_cache.upstream[17]	PVBusSlave
CMN600AE.cmn600_cache.upstream[18]	PVBusSlave
CMN600AE.cmn600_cache.upstream[1]	PVBusSlave
CMN600AE.cmn600_cache.upstream[2]	PVBusSlave
CMN600AE.cmn600_cache.upstream[3]	PVBusSlave
CMN600AE.cmn600_cache.upstream[4]	PVBusSlave
CMN600AE.cmn600_cache.upstream[5]	PVBusSlave
CMN600AE.cmn600_cache.upstream[6]	PVBusSlave
CMN600AE.cmn600_cache.upstream[7]	PVBusSlave
CMN600AE.cmn600_cache.upstream[8]	PVBusSlave
CMN600AE.cmn600_cache.upstream[9]	PVBusSlave
CMN600AE.hnf_exclusive_monitor	PVBusExclusiveMonitor
CMN600AE.hnf_exclusive_monitor.bus_mapper	PVBusMapper

InstanceName	ComponentName
CMN600AE.hni_exclusive_monitor0	PVBusExclusiveMonitor
CMN600AE.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CMN600AE.hni_exclusive_monitor1	PVBusExclusiveMonitor
CMN600AE.hni_exclusive_monitor1.bus_mapper	PVBusMapper
CMN600AE.mpu	PVBusMapper
CMN600AE.ocm_decoder	PVBusMapper
CMN600AE.ocm_exclusive_monitor	PVBusExclusiveMonitor
CMN600AE.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CMN600AE.snf_mapper	PVBusMapper

CMN600AE contains the following CADI targets:

- CMN600AE

About CMN600AE

- For information about the changes in this release, see the Fast Models Portfolio release notes.
- Major IP revisions (rX) are modeled and are controlled by the `revision` parameter.
- Minor IP revisions (pY) are not modeled. There are no functional differences in the model between pY revisions, with the exception of the `periph_id2` register.
- To configure the model, you must have installed Arm® Socrates™. The `mesh_config_file` parameter defines the mesh placement of the CHI nodes. Set it to the name of the yml configuration file emitted by Socrates. You must use version r1p4-01rel0 or later of Socrates to generate the file. For more information about Socrates, see [Arm Socrates](#) on Arm Developer. To download it, contact [Arm Technical Support](#).
- Interconnect models are based on the TRM description and do not typically model RTL defects.

The CMN600AE model supports the following features:

- `rnf`, `rni/rnd`, `hni`, and `snf/sbsx` interface ports. The mapping between the port number and `nodeId` is based on the `nodeId` index. For example, `RNF2` controls `pvbuss_s_rnf[2]`. Its index is specified in the `node_info` register as `logical_id`. Similar behavior can be expected for `HN-I`.

If both `RN-D` and `RN-I` nodes are present, then all starting `rni` ports are mapped to `RN-D` nodes and then the `RN-I` nodes. For example, for CMN600AE with two `RN-D` nodes, one `RN-I` node, and given each `RN-I` or `RN-D` node controls three interface ports, `pvbuss_s_rni[0-2]` maps to `RND0`, `pvbuss_s_rni[3-5]` maps to `RND1` and `pvbuss_s_rni[6-8]` maps to `RNI0`. Similarly, `SN-F` and `SBSX` nodes are mapped to `pvbuss_m_snf[]` ports where starting ports are mapped to `SN-F` and then `SBSX` nodes.

- Mapping `HN*_SLC_SIZE_PARAM` values to cache sizes:
 - -8: OKB when `HN*_SLC_NUM_WAYS_PARAM=16`



This value is not supported by the model.

-
- -2: 128KB when `HN*_SLC_NUM_WAYS_PARAM=16`
 - -1: 256KB when `HN*_SLC_NUM_WAYS_PARAM=16`
 - 0: 512KB when `HN*_SLC_NUM_WAYS_PARAM=16`
 - 1: 1MB when `HN*_SLC_NUM_WAYS_PARAM=16`
 - 2: 2MB when `HN*_SLC_NUM_WAYS_PARAM=16`
 - 3: 3MB when `HN*_SLC_NUM_WAYS_PARAM=12`
 - 3: 4MB when `HN*_SLC_NUM_WAYS_PARAM=16`
 - Maximum number of nodes that have been verified are:
 - 7 RN-Fs
 - 2 RN-Ds
 - 3 RN-Is
 - 1 HN-I
 - 4 HN-Fs
 - 1 SN-F
 - The Address Based Flush (ABF) feature is available. After it has started, the ABF completes instantly. The ABF implementation has the following limitations:
 - In the Fast Model, any HN-F in the interconnect can perform an ABF on any address range even if that range is not part of the System Cache Group (SCG) that the HN-F belongs to.
 - Clearing the OCM with an ABF is not supported. A trace has been added to highlight this.
 - When `HNF_ABF_PR.abf_mode` is Reserved and `HNF_ABF_PR.abf_enable` is set, an error is printed and the ABF status register is set to complete successfully, but no operation takes place.
 - Because an ABF completes instantly, the `SF_PM_TRANSITION_ABORT` state cannot be reached. Contact [Arm Technical Support](#) with questions about support for this error condition.
 - The optional interrupt `INTREQPPU` is not supported.
 - This model implements the following AE-specific features:
 - Memory Protection Unit (MPU), with limitations listed in the Model limitations section.
 - FuSa error logging and reporting using a Fault Management Unit (FMU) and Fault Detection and Control (FDC).
 - Dedicated APB interface into FMU for fault diagnostics and control.

Model limitations

- Out of scope:

- PMU counters are not supported. Counter registers are implemented as RAZ.
- QoS is not supported and all related registers are RAZ/WI.
- Error injection and error generation are not supported. All error registers are RAZ/WI.
- Power, clock, and interrupt signals are not supported.

The P-channel (power) signals `PREQ_LOGIC`, `PSTATE_LOGIC`, `PACCEPT_LOGIC`, `PDENY_LOGIC`, and `PACTIVE_LOGIC`, are not implemented. The model behaves as if `PSTATE_LOGIC` is 5'b00000. So the initial power state of the HN-F nodes is NOSFSLC/OFF. This is reflected in the reset values of `cmn_hns_ppu_pwpr`, which was named `por_hnf_ppu_pwpr` in earlier IP revisions. Software must write the `*_ppu_pwpr` registers to change the power state. Both HAM and FAM power states enable all of the SLC ways.

- Protocol credits and flit buffers are not supported.
- Snoop filtering is not supported.
- Prefetch Target operations are not supported.
- The MTE feature has been minimally tested and the functionality cannot be guaranteed. MTE error injection and detection are not supported.
- An access to a reserved or nonexistent register does not abort. It returns `pv::Tx_Data::TX_OK` and is RAZ/WI.
- Topologies with three I/O masters connected through an RN-I bridge device have been tested. Configurations where multiple I/O masters share a single ACE-Lite port have not been tested.
- RNSAMs external to the mesh network are not functionally supported.



Model parameter `force_rnsam_internal` is true by default. If this parameter is set to false and the topology has an RNFx (non-ESAM), the external bit (bit[31]) of the RNSAM children pointer is enabled. This bit is the only functionality governed by this parameter.

- By default, HN-F hashing uses the address [MAX:12] instead of the actual address [MAX:6], due to the DMI mechanism in the model. Enable the parameter `enable_rnsam_to_hnf_wider_hash` to make hashing logic use the actual address [MAX:6], but this might reduce the simulation speed of the model.
- OKB SLC data RAM and tag RAM are not supported.
- Source-based SLC cache partitioning is not supported.
- Way-based SLC cache partitioning is not supported.
- SLC/LCC partitioning dynamic allocation is not supported.
- Remote chip addresses incorrectly allocate a line in the SLC when `cache_state_modelled=true`. This can cause unexpected cache behavior that does not match the RTL, exhausting the SLC earlier or extra evictions.
- The fields in the `mxp_device_port_connect_info` register corresponding to the number of device credit slices and CAL credit slices are not supported.

- SN-F hashing in HNSAM is not supported. HNSAM SN-F hashing is a single memory range mapped to multiple SN-Fs. Example HNFSAM_Mem_Map:

```
0x00000001000 - 0x00000002000 SNF0 SNF1
```

The model supports mapping HN-F to a single SN-F. Example HNFSAM_Mem_Map:

```
0x00000001000 - 0x00000002000 SNF0
0x00000002000 - 0x00000003000 SNF1
```

- HN-Fs with different SLC sizes in the same configuration are not supported.
- Transactions to addresses unused by Device registers in the Configuration Register Space are always routed to the Configuration Space even when the SAM is configured to use those addresses.
- GIC communication over A4S ports is not supported.
- MPU with data coherency is not supported.
- MPU located in CXRH is not supported.
- No specific Debug Trace (DT) node functionality is supported. See [4. Plug-ins for Fast Models](#) on page 4138 for information about using plug-ins for trace.
- For Power State Transition related flush/disable of the SLC due to HNF_PPU_PWPR register, only the Operational Mode field (bits[7:4]) is considered. The Power Policy field (bits[3:0]) is ignored.
- Setting HAM mode for the Operational Mode field (bits[7:4]) of the HNF_PPU_PWPR is equivalent to FAM mode. The SLCH2 is neither flushed nor disabled according to the Power Policy field (bits[3:0]).
- Only hashing granularities of 4096B and above are supported.
- All RNs are assumed to have the same HN mapping. Programming RNSAMs differently between RNs is not supported.
- RNSAM cannot be programmed with the same HN-F target in a hashed and non-hashed memory region.
- Address bit masking is not supported in RNSAM or HN-F SAM.
- The OCM does not keep track of the total memory used across the S/NS security world and does not perform any eviction either. It is up to the software to make sure it does not oversubscribe the OCM.
- There is no support for RAS.
- The following limitations apply to System Cache Groups and Hash Target Groups:
 - A mix of local and remote HN-F targets are not supported.
 - CCG/CXRH target IDs in the HN target ID table are not supported (`sys_cache_grp_hn_nodeid_reg`).
 - The `hashed_target_grp_hnp_nodeid_reg` which is used for CCG/HN-P target IDs is not supported.
 - System/Hash Target Groups only support HN-Fs.

- AXID hashing across HN-P/CCGs is not supported.

About the `debug_force_snoop` parameter

The CMN interconnect normally starts up with snooping disabled.

The parameters `rnf_sci_enable` and `rni_sci_enable` determine which connections are managed by the System Coherency Interface. Connections that are managed by the System Coherency Interface look after themselves for entering and leaving the coherency domain and `debug_force_snoop` has no effect on those ports.

However, for connections that are not managed by the System Coherency Interface, `debug_force_snoop` enables the system upstream of a port to be snooped if the upstream is not in reset and if `acchannelen_rnf` allows it.

Therefore, software does not need to know how to turn snoops in the interconnect on and off. The reset state of the upstream `rnf` ports is determined by the `rnf_upstream_reset_state[]` signals and must be connected for any port that could go into reset and you want managed by `debug_force_snoop`.

If the connection is managed by SCI or can never go into reset then you need not connect this and the interconnect assumes that the upstream system is not in reset and will always enable the snoops if `ACCHANNELENSx` allows it.

If you fail to connect `rnf_upstream_reset_state[]` correctly, then the upstream system might receive snoop messages while in reset and it will complain that it received a snoop request while it was in reset.

If software overrides the enables then the next change to `rnf_upstream_reset_state[]` or the interconnect reset will overwrite the software values.

Using this operation along with software controlling the enables could confuse the software. This option does not remove the responsibility of the upstream system to clean any shared dirty data from its caches before going into reset.



This parameter is for debug purposes only. Do not use it instead of correctly configuring CMN.

Ports for CMN600AE

Table 3-879: Ports

Name	Protocol	Type	Description
<code>event_clusters</code>	Signal	Peer	CPU event communication signal from the clusters.
<code>event_downstream_link_signal[2]</code>	Signal	Master	CPU event communication signal to the CMLHub. Event from the CMN towards the Hub NOTE that these are virtual "links" on underlying PCIe hardware bus.

Name	Protocol	Type	Description
event_upstream_link_signal[2]	Signal	Slave	Event from the Hub towards the CMN
fmu_eri	Signal	Master	FMU signal for critical errors
fmu_fhi	Signal	Master	FMU signal for non-critical errors
pvbus_m_cml	PVBus	Master	CML downstream ports
pvbus_m_cml_cfg	PVBus	Master	CML downstream hub configuration port
pvbus_m_hni[4]	PVBus	Master	HNI downstream ports.
pvbus_m_snf[4]	PVBus	Master	SNF downstream port.
pvbus_s_apb	PVBus	Slave	APB interface port.
pvbus_s_cml	PVBus	Slave	CML upstream ports
pvbus_s_rnf[8]	PVBus	Slave	RNF upstream ports.
pvbus_s_rni[24]	PVBus	Slave	RNI upstream ports.
reset_in	Signal	Slave	Reset signal.
rnf_sci_s[8]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-F in and out of the coherency domain.
rnf_upstream_reset_in[8]	Signal	Slave	Used by the interconnect to determine the reset state of the RNF manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.
rni_sci_s[24]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-D in and out of the coherency domain.
rni_upstream_reset_in[24]	Signal	Slave	Used by the interconnect to determine the reset state of the RNI manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.

Parameters for CMN600AE

acchannelen_rnf

DEPRECATED: Will be removed after FM 11.18. Use 'rnf_sci_enable' instead. For each rnf port, indicates if the port is populated with a snoop responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

acchannelen_rni

DEPRECATED: Will be removed after FM 11.18. Use 'rni_sci_enable' instead. For each rni port, indicates if the port is populated with a dvm responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x0

debug_force_snoop

The CMN600AE interconnect will normally start with snooping disabled. The parameter `rnf_sci_enable` and `rni_sci_enable` determines which connections are managed by the System Coherency Interface. For connections that are managed by the System Coherency Interface, then they will look after themselves for entering and leaving the coherency domain and this parameter has no effect for those ports. However, for connections that are `_not_` managed by the System Coherency Interface, this parameter enables the upstream system of a port to be snooped if the upstream is not in reset and if `acchannelen_rnf` allows it. NOTE: This parameter is for debug purpose only Do not use this parameter instead of correctly configuring CMN. Refer CMN600AE documentation page for more info about this parameter in fast models reference manual

Type

bool

Default value

0x0

disable_CML_port

If true the model won't connect the CML port when there are nodes which support the CML feature (ie CXG nodes).

Type

bool

Default value

0x0

enable_logger

Enable PVBUSLoggers for the downstream ports in the CMN model.

Type

bool

Default value

0x0

enable_rnsam_to_hnf_wider_hash

Enable support of wider hash for the RNSAM to HNF communication. If this variable enabled, then bits[47:6] from PA used in hashing function. By default it is [47:12].

Type

bool

Default value

0x0

fdc_key

Por_fdc_key register value is checked against this key.

Type

int

Default value

0x0

fmu_key

Por_fmu_key register value is checked against this key.

Type

int

Default value

0x0

force_rnsam_internal

Force all RNSAMs to be internal independently of the mesh topology.

Type

bool

Default value

0x1

mesh_config_file

Name of a file containing mesh placement of CMN600AE components.

Type

string

Default value/tmp/plgbuild/abs_build/1378878_61223/trunk/work/fastsim/autodocs/..//RegTests/
Interconnect/CMN600AE/cmn600ae_all_rnfs_on_port_0_or_1.yml**number_of_mpu_programmable_regions**

Number of MPU programmable regions. Valid values are 0, 8, 16 and 32.

Type

int

Default value

0x20

periphbase

Value for PERIPHBASE. Bits [25:0] are treated 0

Type

int

Default value

0x20000000

print_cmn_ccix_config

Print information about the CCIX configuration.

Type

bool

Default value

0x0

print_cmn_config

Print the mesh topology and children pointers acquired from the YML file.

Type

bool

Default value

0x0

revision

Component revision. Currently supports r1p0.

Type

string

Default value

r1p0

rnf_sci_enable

For each rnf port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

rni_sci_enable

For each rni port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

show_banner

Show component banner: 0=>supress entire banner, 1=>suppress config file, 2+=>show full banner.

Type

int

Default value

0x2

skip_cmnn_config_check

Skip any topology configuration checks. The maximum number of devices per type not verified.

Type

bool

Default value

0x0

use_yaml_periphbase

Use yaml param CFGM_PERIPHBASE_PARAM to specify periphbase address. If false, model parameter 'periphbase' will be used.

Type

bool

Default value

0x0

3.10.15 CMN600CMLHub

CMN600 CML Interconnect Hub Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-880: IP revisions support

Revision	Quality level
N/A	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CMN600CMLHub

This model has the following Iris instances:

Table 3-881: CMN600CMLHub Iris instances

InstanceName	ComponentName
CMN600CMLHub	CMN600CMLHub
CMN600CMLHub.CMN600CMLHubCache	PVCache
CMN600CMLHub.CMN600CMLHubCache.upstream[0]	PVBusSlave
CMN600CMLHub.CMN600CMLHubCache.upstream[1]	PVBusSlave
CMN600CMLHub.CMN600CMLHubCache.upstream[2]	PVBusSlave
CMN600CMLHub.CMN600CMLHubCache.upstream[3]	PVBusSlave
CMN600CMLHub.bus_s_cfg	PVBusSlave
CMN600CMLHub.cache_downstream_exclusive_monitor0	PVBusExclusiveMonitor
CMN600CMLHub.cache_downstream_exclusive_monitor0.bus_mapper	PVBusMapper
CMN600CMLHub.cache_downstream_exclusive_monitor1	PVBusExclusiveMonitor
CMN600CMLHub.cache_downstream_exclusive_monitor1.bus_mapper	PVBusMapper
CMN600CMLHub.cache_downstream_exclusive_monitor2	PVBusExclusiveMonitor
CMN600CMLHub.cache_downstream_exclusive_monitor2.bus_mapper	PVBusMapper
CMN600CMLHub.cache_downstream_exclusive_monitor3	PVBusExclusiveMonitor
CMN600CMLHub.cache_downstream_exclusive_monitor3.bus_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-882: CMN600CMLHub MTI instances

InstanceName	ComponentName
CMN600CMLHub	CMN600CML
CMN600CMLHub.CMN600CMLHubCache	PVCache
CMN600CMLHub.CMN600CMLHubCache.upstream[0]	PVBusSlave
CMN600CMLHub.CMN600CMLHubCache.upstream[1]	PVBusSlave
CMN600CMLHub.CMN600CMLHubCache.upstream[2]	PVBusSlave
CMN600CMLHub.CMN600CMLHubCache.upstream[3]	PVBusSlave
CMN600CMLHub.bus_s_cfg	PVBusSlave
CMN600CMLHub.cache_downstream_exclusive_monitor0	PVBusExclusiveMonitor
CMN600CMLHub.cache_downstream_exclusive_monitor0.bus_mapper	PVBusMapper
CMN600CMLHub.cache_downstream_exclusive_monitor1	PVBusExclusiveMonitor
CMN600CMLHub.cache_downstream_exclusive_monitor1.bus_mapper	PVBusMapper
CMN600CMLHub.cache_downstream_exclusive_monitor2	PVBusExclusiveMonitor
CMN600CMLHub.cache_downstream_exclusive_monitor2.bus_mapper	PVBusMapper
CMN600CMLHub.cache_downstream_exclusive_monitor3	PVBusExclusiveMonitor
CMN600CMLHub.cache_downstream_exclusive_monitor3.bus_mapper	PVBusMapper

CMN600CMLHub contains the following CADI targets:

- CMN600CMLHub

Ports for CMN600CMLHub

Table 3-883: Ports

Name	Protocol	Type	Description
event_downstream_link_signal[4]	Signal	Slave	CPU downstream event communication signal.
event_upstream_link_signal[4]	Signal	Master	CPU upstream event communication signal.
pvbus_m[4]	PVBus	Master	Downstream CCIX port.
pvbus_s[4]	PVBus	Slave	Upstream CCIX ports.
pvbus_s_cfg[4]	PVBus	Slave	Upstream config ports.
reset_signal	Signal	Slave	Reset signal.

Parameters for CMN600CMLHub

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x0

enable_logger

Enable PVBusLoggers for the downstream ports in the CMN model.

Type

bool

Default value

0x0

3.10.16 CMN650

CMN650 Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-884: IP revisions support

Revision	Quality level
r1p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CMN650

This model has the following Iris instances:

Table 3-885: CMN650 Iris instances

InstanceName	ComponentName
CMN650	CMN650
CMN650.bus_slave_ocm_NS	PVBusSlave
CMN650.bus_slave_ocm_S	PVBusSlave
CMN650.cmn600_cache	PVCache
CMN650.cmn600_cache.upstream[0]	PVBusSlave
CMN650.cmn600_cache.upstream[10]	PVBusSlave
CMN650.cmn600_cache.upstream[11]	PVBusSlave
CMN650.cmn600_cache.upstream[12]	PVBusSlave
CMN650.cmn600_cache.upstream[13]	PVBusSlave
CMN650.cmn600_cache.upstream[14]	PVBusSlave
CMN650.cmn600_cache.upstream[15]	PVBusSlave
CMN650.cmn600_cache.upstream[16]	PVBusSlave
CMN650.cmn600_cache.upstream[17]	PVBusSlave
CMN650.cmn600_cache.upstream[18]	PVBusSlave
CMN650.cmn600_cache.upstream[19]	PVBusSlave
CMN650.cmn600_cache.upstream[1]	PVBusSlave
CMN650.cmn600_cache.upstream[20]	PVBusSlave
CMN650.cmn600_cache.upstream[21]	PVBusSlave
CMN650.cmn600_cache.upstream[22]	PVBusSlave
CMN650.cmn600_cache.upstream[23]	PVBusSlave
CMN650.cmn600_cache.upstream[24]	PVBusSlave
CMN650.cmn600_cache.upstream[25]	PVBusSlave
CMN650.cmn600_cache.upstream[26]	PVBusSlave
CMN650.cmn600_cache.upstream[27]	PVBusSlave
CMN650.cmn600_cache.upstream[28]	PVBusSlave
CMN650.cmn600_cache.upstream[29]	PVBusSlave
CMN650.cmn600_cache.upstream[2]	PVBusSlave
CMN650.cmn600_cache.upstream[30]	PVBusSlave
CMN650.cmn600_cache.upstream[31]	PVBusSlave
CMN650.cmn600_cache.upstream[32]	PVBusSlave
CMN650.cmn600_cache.upstream[33]	PVBusSlave
CMN650.cmn600_cache.upstream[3]	PVBusSlave
CMN650.cmn600_cache.upstream[4]	PVBusSlave
CMN650.cmn600_cache.upstream[5]	PVBusSlave
CMN650.cmn600_cache.upstream[6]	PVBusSlave

InstanceName	ComponentName
CMN650.cmn600_cache.upstream[7]	PVBusSlave
CMN650.cmn600_cache.upstream[8]	PVBusSlave
CMN650.cmn600_cache.upstream[9]	PVBusSlave
CMN650.hnf_exclusive_monitor	PVBusExclusiveMonitor
CMN650.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CMN650.hni_exclusive_monitor0	PVBusExclusiveMonitor
CMN650.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CMN650.hni_exclusive_monitor1	PVBusExclusiveMonitor
CMN650.hni_exclusive_monitor1.bus_mapper	PVBusMapper
CMN650.hni_exclusive_monitor2	PVBusExclusiveMonitor
CMN650.hni_exclusive_monitor2.bus_mapper	PVBusMapper
CMN650.ocm_decoder	PVBusMapper
CMN650.ocm_exclusive_monitor	PVBusExclusiveMonitor
CMN650.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CMN650.snf_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-886: CMN650 MTI instances

InstanceName	ComponentName
CMN650	CMN650
CMN650.bus_slave_ocm_NS	PVBusSlave
CMN650.bus_slave_ocm_S	PVBusSlave
CMN650.cmn600_cache	CMN600Cache
CMN650.cmn600_cache.upstream[0]	PVBusSlave
CMN650.cmn600_cache.upstream[10]	PVBusSlave
CMN650.cmn600_cache.upstream[11]	PVBusSlave
CMN650.cmn600_cache.upstream[12]	PVBusSlave
CMN650.cmn600_cache.upstream[13]	PVBusSlave
CMN650.cmn600_cache.upstream[14]	PVBusSlave
CMN650.cmn600_cache.upstream[15]	PVBusSlave
CMN650.cmn600_cache.upstream[16]	PVBusSlave
CMN650.cmn600_cache.upstream[17]	PVBusSlave
CMN650.cmn600_cache.upstream[18]	PVBusSlave
CMN650.cmn600_cache.upstream[19]	PVBusSlave
CMN650.cmn600_cache.upstream[1]	PVBusSlave
CMN650.cmn600_cache.upstream[20]	PVBusSlave
CMN650.cmn600_cache.upstream[21]	PVBusSlave
CMN650.cmn600_cache.upstream[22]	PVBusSlave
CMN650.cmn600_cache.upstream[23]	PVBusSlave

InstanceName	ComponentName
CMN650.cmn600_cache.upstream[24]	PVBusSlave
CMN650.cmn600_cache.upstream[25]	PVBusSlave
CMN650.cmn600_cache.upstream[26]	PVBusSlave
CMN650.cmn600_cache.upstream[27]	PVBusSlave
CMN650.cmn600_cache.upstream[28]	PVBusSlave
CMN650.cmn600_cache.upstream[29]	PVBusSlave
CMN650.cmn600_cache.upstream[2]	PVBusSlave
CMN650.cmn600_cache.upstream[30]	PVBusSlave
CMN650.cmn600_cache.upstream[31]	PVBusSlave
CMN650.cmn600_cache.upstream[32]	PVBusSlave
CMN650.cmn600_cache.upstream[33]	PVBusSlave
CMN650.cmn600_cache.upstream[3]	PVBusSlave
CMN650.cmn600_cache.upstream[4]	PVBusSlave
CMN650.cmn600_cache.upstream[5]	PVBusSlave
CMN650.cmn600_cache.upstream[6]	PVBusSlave
CMN650.cmn600_cache.upstream[7]	PVBusSlave
CMN650.cmn600_cache.upstream[8]	PVBusSlave
CMN650.cmn600_cache.upstream[9]	PVBusSlave
CMN650.hnf_exclusive_monitor	PVBusExclusiveMonitor
CMN650.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CMN650.hni_exclusive_monitor0	PVBusExclusiveMonitor
CMN650.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CMN650.hni_exclusive_monitor1	PVBusExclusiveMonitor
CMN650.hni_exclusive_monitor1.bus_mapper	PVBusMapper
CMN650.hni_exclusive_monitor2	PVBusExclusiveMonitor
CMN650.hni_exclusive_monitor2.bus_mapper	PVBusMapper
CMN650.ocm_decoder	PVBusMapper
CMN650.ocm_exclusive_monitor	PVBusExclusiveMonitor
CMN650.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CMN650.snf_mapper	PVBusMapper

CMN650 contains the following CADI targets:

- CMN650

About CMN650

- For information about the changes in this release, see the Fast Models Portfolio release notes.
- Major IP revisions (rX) are modeled and are controlled by the `revision` parameter.
- Minor IP revisions (pY) are not modeled. There are no functional differences in the model between pY revisions, with the exception of the `periph_id2` register.

- To configure the model, you must have installed Arm® Socrates™. The `mesh_config_file` parameter defines the mesh placement of the CHI nodes. Set it to the name of the yml configuration file emitted by Socrates. You must use version r1p4-01rel0 or later of Socrates to generate the file. For more information about Socrates, see [Arm Socrates](#) on Arm Developer. To download it, contact [Arm Technical Support](#).
- Interconnect models are based on the TRM description and do not typically model RTL defects.

The model supports the following features:

- `rnf`, `rni/rnd`, `hni`, and `snf/sbsx` interface ports. The mapping between the port number and `nodeId` is based on the `nodeId` index. For example, `RNF2` controls `pvbuss_s_rnf[2]`. Its index is specified in the `node_info` register as `logical_id`. Similar behavior can be expected for `HN-I`.

If both `RN-D` and `RN-I` nodes are present, then all starting `rni` ports are mapped to `RN-D` nodes and then the `RN-I` nodes. For example, for `CMN650` with two `RN-D` nodes, one `RN-I` node, and given each `RN-I` or `RN-D` node controls three interface ports, `pvbuss_s_rni[0-2]` maps to `RND0`, `pvbuss_s_rni[3-5]` maps to `RND1` and `pvbuss_s_rni[6-8]` maps to `RNI0`.

Similarly, `SN-F` and `SBSX` nodes are mapped to `pvbuss_m_snf[]` ports where starting ports are mapped to `SN-F` and then `SBSX` nodes.

- Mapping `HN*_SLC_SIZE_PARAM` values to cache sizes:
 - -8: OKB where `HN*_SLC_NUM_WAYS_PARAM=16`.



This value is not supported by the model.

-
- -2: 128KB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - -1: 256KB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - 0: 512KB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - 1: 1MB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - 2: 2MB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - 3: 3MB where `HN*_SLC_NUM_WAYS_PARAM=12`
 - 3: 4MB where `HN*_SLC_NUM_WAYS_PARAM=16`
 - The Address Based Flush (ABF) feature is available. After it has started, the ABF completes instantly. The ABF implementation has the following limitations:
 - In the Fast Model, any `HN-F` in the interconnect can perform an ABF on any address range even if that range is not part of the System Cache Group (SCG) that the `HN-F` belongs to.
 - Clearing the OCM with an ABF is not supported. A trace has been added to highlight this.
 - When `HNF_ABF_PR.abf_mode` is `Reserved` and `HNF_ABF_PR.abf_enable` is set, an error is printed and the ABF status register is set to complete successfully, but no operation takes place.

- Because an ABF completes instantly, the `SF_PM_TRANSITION_ABORT` state cannot be reached. Contact [Arm Technical Support](#) with questions about support for this error condition.
- The optional interrupt `INTREQPPU` is not supported.

Model limitations

- Out of scope:
 - PMU counters are not supported. Counter registers are implemented as RAZ.
 - QoS is not supported and all related registers are RAZ/WI.
 - Error injection and error generation are not supported. All error registers are RAZ/WI.
 - Power, clock, and interrupt signals are not supported.

The P-channel (power) signals `PREQ_LOGIC`, `PSTATE_LOGIC`, `PACCEPT_LOGIC`, `PDENY_LOGIC`, and `PACTIVE_LOGIC`, are not implemented. The model behaves as if `PSTATE_LOGIC` is 5'b00000. So the initial power state of the HN-F nodes is NOSFSLC/OFF. This is reflected in the reset values of `cmn_hns_ppu_pwpr`, which was named `por_hnf_ppu_pwpr` in earlier IP revisions. Software must write the `*_ppu_pwpr` registers to change the power state. Both HAM and FAM power states enable all of the SLC ways.

- Protocol credits and flit buffers are not supported.
- Snoop filtering is not supported.
- Prefetch Target operations are not supported.
- The MTE feature has been minimally tested and the functionality cannot be guaranteed. MTE error injection and detection are not supported.
- An access to a reserved or nonexistent register does not abort. It returns `pv::Tx_Data::TX_OK` and is RAZ/WI.
- Topologies with three I/O masters connected through an RN-I bridge device have been tested. Configurations where multiple I/O masters share a single ACE-lite port have not been tested.
- RNSAMs external to the mesh network are not functionally supported.



Model parameter `force_rnsam_internal` is true by default. If this parameter is set to false and the topology has an RNFx (non-ESAM), the external bit (bit[31]) of the RNSAM children pointer is enabled. This bit is the only functionality governed by this parameter.

- By default, HN-F hashing uses the address [MAX:12] instead of the actual address [MAX:6], due to the DMI mechanism in the model. Enable the parameter `enable_rnsam_to_hnf_wider_hash` to make hashing logic use the actual address [MAX:6], but this might reduce the simulation speed of the model.
- OKB SLC data RAM and tag RAM are not supported.
- Source-based SLC cache partitioning is not supported.
- Way-based SLC cache partitioning is not supported.
- SLC/LCC partitioning dynamic allocation is not supported.

- Remote chip addresses incorrectly allocate a line in the SLC when `cache_state_modelled=true`. This can cause unexpected cache behavior that does not match the RTL, exhausting the SLC earlier or extra evictions.
- The fields in the `mvp_device_port_connect_info` register corresponding to the number of device credit slices and the number of CAL credit slices are not supported.
- SN-F hashing in HNSAM is not supported. HNSAM SN-F hashing is a single memory range mapped to multiple SN-Fs. Example HNFSAM_Mem_Map:

```
0x00000001000 - 0x0000002000 SNF0 SNF1
```

The model supports mapping HN-F to a single SN-F. Example HNFSAM_Mem_Map:

```
0x00000001000 - 0x0000002000 SNF0
0x0000002000 - 0x0000003000 SNF1
```

- HN-Fs with different SLC sizes in the same configuration are not supported.
- Transactions to addresses unused by Device registers in the Configuration Register Space are always routed to the Configuration Space even when the SAM is configured to use those addresses.
- GIC communication over A4S ports is not supported.
- MPAM features are not supported in the model, and the configuration-dependent register reset values do not match the RTL. Some of the registers (`por_hnf_mpam_idr` to `por_hnf_mpam_mbwumon_idr`) shared between security modes are not present in MPAM_S.
- The following limitations apply to System Cache Groups and Hash Target Groups:
 - A mix of local and remote HN-F targets is not supported.
 - CCG/CXRH target IDs in the HN target ID table are not supported (`sys_cache_grp_hn_nodeid_reg`).
 - The `hashed_target_grp_hnp_nodeid_reg` which is used for CCG/HN-P target IDs is not supported.
 - System/Hash Target Groups only support HN-Fs.
 - AXID hashing across HN-P/CCGs is not supported.
- The MPAM reset values described in `por_hnf_mpam_ns_por_hnf_mpam_idr[31:24]` default to the not-supported state. The model parameter `hnf_mpam_idr_override` can be used to override this but does not implement any of the functionality.
- MPAM_S `secure_register_groups_override` is not supported.
- No specific Debug Trace (DT) node functionality is supported. See [4. Plug-ins for Fast Models](#) on page 4138 for information about using plug-ins for trace.
- For Power State Transition related flush/disable of the SLC due to HNF_PPU_PWPR register, only the Operational Mode field (bits[7:4]) is considered. The Power Policy field (bits[3:0]) is ignored.
- Setting HAM mode for the Operational Mode field (bits[7:4]) of the HNF_PPU_PWPR is equivalent to FAM mode. The SLCH2 is neither flushed nor disabled according to the Power Policy field (bits[3:0]).

- Only hashing granularities of 4096B and above are supported.
- The OCM does not keep track of the total memory used across the S/NS security world and does not perform any eviction either. It is up to the software to make sure it does not oversubscribe the OCM.
- All RNs are assumed to have the same HN mapping. Programming RNSAMs differently between RNs is not supported.
- RNSAM cannot be programmed with the same HN-F target in a hashed and non-hashed memory region.
- `RNSAM_STATUS.default_target_type` and `RNSAM_STATUS.default_nodeid` are used as the default target only when `RNSAM_STATUS.use_default_node=1`. Otherwise, when a txn is sent into the model that does not belong to any of the address regions programmed in the RNSAM, it is routed to the HN-D irrespective of what is programmed in the `RNSAM_STATUS.default_target_type` and `RNSAM_STATUS.default_nodeid` fields.
- Address bit masking is not supported in RNSAM or HN-F SAM.
- HN-F SAM limitations:
 - Address masking in default hash regions in HN-F SAM is not supported.
 - Hashing in default hash regions in HN-F SAM is not supported. However, trace `SNF_HASHING_Target_SNF` displays the target SNF as if 3SN/6SN hashing were enabled.



The transaction is always routed to the SN0 nodeid programmed in the SAM_CONTROL register.

- Re-programming regions in HN-F SAM is not tested.
- Hashing across CCGs in HN-F SAM is not supported.
- For an HTG/SCG, the model does not consider the secondary region if the primary region is not valid.
- There is no support for RAS.

About the `debug_force_snoop` parameter

The CMN interconnect normally starts up with snooping disabled.

The parameters `rnf_sci_enable` and `rni_sci_enable` determine which connections are managed by the System Coherency Interface. Connections that are managed by the System Coherency Interface look after themselves for entering and leaving the coherency domain and `debug_force_snoop` has no effect on those ports.

However, for connections that are not managed by the System Coherency Interface, `debug_force_snoop` enables the system upstream of a port to be snooped if the upstream is not in reset and if `acchannelen_rnf` allows it.

Therefore, software does not need to know how to turn snoops in the interconnect on and off. The reset state of the upstream rnf ports is determined by the `rnf_upstream_reset_state[]`

signals and must be connected for any port that could go into reset and you want managed by `debug_force_snoop`.

If the connection is managed by SCI or can never go into reset then you need not connect this and the interconnect assumes that the upstream system is not in reset and will always enable the snoops if `ACCHANNELENSx` allows it.

If you fail to connect `rnf_upstream_reset_state[]` correctly, then the upstream system might receive snoop messages while in reset and it will complain that it received a snoop request while it was in reset.

If software overrides the enables then the next change to `rnf_upstream_reset_state[]` or the interconnect reset will overwrite the software values.

Using this operation along with software controlling the enables could confuse the software. This option does not remove the responsibility of the upstream system to clean any shared dirty data from its caches before going into reset.



This parameter is for debug purposes only. Do not use it instead of correctly configuring CMN.

Ports for CMN650

Table 3-887: Ports

Name	Protocol	Type	Description
<code>event_clusters</code>	Signal	Peer	CPU event communication signal from the clusters.
<code>event_downstream_link_signal[10]</code>	Signal	Master	CPU event communication signal to the CMLHub. Event from the CMN towards the Hub NOTE that these are virtual "links" on underlying PCIe hardware bus.
<code>event_upstream_link_signal[10]</code>	Signal	Slave	Event from the Hub towards the CMN
<code>pvbuse_m_cml</code>	PVBus	Master	CML downstream ports
<code>pvbuse_m_cml_cfg</code>	PVBus	Master	CML downstream hub configuration port
<code>pvbuse_m_hni[16]</code>	PVBus	Master	HNI downstream ports.
<code>pvbuse_m_snf[40]</code>	PVBus	Master	SNF downstream port.
<code>pvbuse_s_apb</code>	PVBus	Slave	APB interface port.
<code>pvbuse_s_cml</code>	PVBus	Slave	CML upstream ports
<code>pvbuse_s_rnf[64]</code>	PVBus	Slave	RNF upstream ports.
<code>pvbuse_s_rni[96]</code>	PVBus	Slave	RNI upstream ports.
<code>reset_in</code>	Signal	Slave	Reset signal.
<code>rnf_sci_s[64]</code>	SystemCoherencyInterface	Slave	System Coherency port to move the RN-F in and out of the coherency domain.

Name	Protocol	Type	Description
rnf_upstream_reset_in[64]	Signal	Slave	Used by the interconnect to determine the reset state of the RNF manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.
rni_sci_s[96]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-D in and out of the coherency domain.
rni_upstream_reset_in[96]	Signal	Slave	Used by the interconnect to determine the reset state of the RNI manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.

Parameters for CMN650

acchannelen_rnf

DEPRECATED: Will be removed after FM 11.18. Use 'rnf_sci_enable' instead. For each rnf port, indicates if the port is populated with a snoop responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

acchannelen_rni

DEPRECATED: Will be removed after FM 11.18. Use 'rni_sci_enable' instead. For each rni port, indicates if the port is populated with a dvm responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x0

debug_force_snoop

The CMN650 interconnect will normally start with snooping disabled. The parameter rnf_sci_enable and rni_sci_enable determines which connections are managed by the System Coherency Interface. For connections that are managed by the System Coherency Interface, then they will look after themselves for entering and leaving the coherency domain and this parameter has no effect for those ports. However, for connections that are *_not_* managed by the System Coherency Interface, this parameter enables the upstream system of a port to be snooped if the upstream is not in reset and if acchannelen_rnf allows it. NOTE:

This parameter is for debug purpose only Do not use this parameter instead of correctly configuring CMN. Refer CMN650 documentation page for more info about this parameter in fast models reference manual

Type

bool

Default value

0x0

disable_CML_port

If true the model won't connect the CML port when there are nodes which support the CML feature (ie CXG nodes).

Type

bool

Default value

0x0

enable_logger

Enable PVBUSLoggers for the downstream ports in the CMN model.

Type

bool

Default value

0x0

enable_rnsam_to_hnf_wider_hash

Enable support of wider hash for the RNSAM to HNF communication. If this variable enabled, then bits[47:6] from PA used in hashing function. By default it is [47:12].

Type

bool

Default value

0x0

force_rnsam_internal

Force all RNSAMs to be internal independently of the mesh topology.

Type

bool

Default value

0x1

hnf_mpam_idr_override

Set to override hnf_mpam_idr[31:24] value. Bit[28] is Reserved and is ignored.

Type

int

Default value

0x0

mesh_config_file

Name of a file containing mesh placement of CMN650 components.

Type

string

Default value/tmp/plgbuild/abs_build/1378878_61223/trunk/work/fastsim/autodocs/../../Platforms/
LISA/CSS/Kits4/cmn650_rdv1.yml**periphbase**

Value for PERIPHBASE. Bits [27:0] are treated 0

Type

int

Default value

0x20000000

print_cmn_ccix_config

Print information about the CCIX configuration.

Type

bool

Default value

0x0

print_cmn_config

Print the mesh topology and children pointers acquired from the YAML file.

Type

bool

Default value

0x0

revision

Component revision. Currently supports r1p1.

Type

string

Default value

r1p1

rnf_sci_enable

For each rnf port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

rni_sci_enable

For each rni port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

show_banner

Show component banner: 0=>supress entire banner, 1=>suppress config file, 2+=>show full banner.

Type

int

Default value

0x2

skip_cmn_config_check

Skip any topology configuration checks. The maximum number of devices per type not verified.

Type

bool

Default value

0x0

use_yaml_periphbase

Use yaml param CFGM_PERIPHBASE_PARAM to specify periphbase address. If false, model parameter 'periphbase' will be used.

Type

bool

Default value

0x0

3.10.17 CMN650R2

CMN650R2 Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-888: IP revisions support

Revision	Quality level
r2p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for CMN650R2

This model has the following Iris instances:

Table 3-889: CMN650R2 Iris instances

InstanceName	ComponentName
CMN650R2	CMN650R2
CMN650R2.bus_slave_ocm_NS	PVBusSlave
CMN650R2.bus_slave_ocm_S	PVBusSlave
CMN650R2.cmn600_cache	PVCache
CMN650R2.cmn600_cache.upstream[0]	PVBusSlave
CMN650R2.cmn600_cache.upstream[10]	PVBusSlave
CMN650R2.cmn600_cache.upstream[11]	PVBusSlave
CMN650R2.cmn600_cache.upstream[12]	PVBusSlave
CMN650R2.cmn600_cache.upstream[13]	PVBusSlave
CMN650R2.cmn600_cache.upstream[14]	PVBusSlave
CMN650R2.cmn600_cache.upstream[15]	PVBusSlave
CMN650R2.cmn600_cache.upstream[16]	PVBusSlave
CMN650R2.cmn600_cache.upstream[17]	PVBusSlave
CMN650R2.cmn600_cache.upstream[18]	PVBusSlave
CMN650R2.cmn600_cache.upstream[19]	PVBusSlave
CMN650R2.cmn600_cache.upstream[1]	PVBusSlave
CMN650R2.cmn600_cache.upstream[20]	PVBusSlave
CMN650R2.cmn600_cache.upstream[21]	PVBusSlave
CMN650R2.cmn600_cache.upstream[22]	PVBusSlave
CMN650R2.cmn600_cache.upstream[23]	PVBusSlave
CMN650R2.cmn600_cache.upstream[24]	PVBusSlave
CMN650R2.cmn600_cache.upstream[25]	PVBusSlave
CMN650R2.cmn600_cache.upstream[26]	PVBusSlave
CMN650R2.cmn600_cache.upstream[27]	PVBusSlave

InstanceName	ComponentName
CMN650R2.cmn600_cache.upstream[28]	PVBusSlave
CMN650R2.cmn600_cache.upstream[29]	PVBusSlave
CMN650R2.cmn600_cache.upstream[2]	PVBusSlave
CMN650R2.cmn600_cache.upstream[30]	PVBusSlave
CMN650R2.cmn600_cache.upstream[31]	PVBusSlave
CMN650R2.cmn600_cache.upstream[32]	PVBusSlave
CMN650R2.cmn600_cache.upstream[33]	PVBusSlave
CMN650R2.cmn600_cache.upstream[3]	PVBusSlave
CMN650R2.cmn600_cache.upstream[4]	PVBusSlave
CMN650R2.cmn600_cache.upstream[5]	PVBusSlave
CMN650R2.cmn600_cache.upstream[6]	PVBusSlave
CMN650R2.cmn600_cache.upstream[7]	PVBusSlave
CMN650R2.cmn600_cache.upstream[8]	PVBusSlave
CMN650R2.cmn600_cache.upstream[9]	PVBusSlave
CMN650R2.hnf_exclusive_monitor	PVBusExclusiveMonitor
CMN650R2.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CMN650R2.hni_exclusive_monitor0	PVBusExclusiveMonitor
CMN650R2.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CMN650R2.hni_exclusive_monitor1	PVBusExclusiveMonitor
CMN650R2.hni_exclusive_monitor1.bus_mapper	PVBusMapper
CMN650R2.hni_exclusive_monitor2	PVBusExclusiveMonitor
CMN650R2.hni_exclusive_monitor2.bus_mapper	PVBusMapper
CMN650R2.ocm_decoder	PVBusMapper
CMN650R2.ocm_exclusive_monitor	PVBusExclusiveMonitor
CMN650R2.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CMN650R2.snf_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-890: CMN650R2 MTI instances

InstanceName	ComponentName
CMN650R2	CMN650R2
CMN650R2.bus_slave_ocm_NS	PVBusSlave
CMN650R2.bus_slave_ocm_S	PVBusSlave
CMN650R2.cmn600_cache	CMN600Cache
CMN650R2.cmn600_cache.upstream[0]	PVBusSlave
CMN650R2.cmn600_cache.upstream[10]	PVBusSlave
CMN650R2.cmn600_cache.upstream[11]	PVBusSlave
CMN650R2.cmn600_cache.upstream[12]	PVBusSlave
CMN650R2.cmn600_cache.upstream[13]	PVBusSlave

InstanceName	ComponentName
CMN650R2.cmn600_cache.upstream[14]	PVBusSlave
CMN650R2.cmn600_cache.upstream[15]	PVBusSlave
CMN650R2.cmn600_cache.upstream[16]	PVBusSlave
CMN650R2.cmn600_cache.upstream[17]	PVBusSlave
CMN650R2.cmn600_cache.upstream[18]	PVBusSlave
CMN650R2.cmn600_cache.upstream[19]	PVBusSlave
CMN650R2.cmn600_cache.upstream[1]	PVBusSlave
CMN650R2.cmn600_cache.upstream[20]	PVBusSlave
CMN650R2.cmn600_cache.upstream[21]	PVBusSlave
CMN650R2.cmn600_cache.upstream[22]	PVBusSlave
CMN650R2.cmn600_cache.upstream[23]	PVBusSlave
CMN650R2.cmn600_cache.upstream[24]	PVBusSlave
CMN650R2.cmn600_cache.upstream[25]	PVBusSlave
CMN650R2.cmn600_cache.upstream[26]	PVBusSlave
CMN650R2.cmn600_cache.upstream[27]	PVBusSlave
CMN650R2.cmn600_cache.upstream[28]	PVBusSlave
CMN650R2.cmn600_cache.upstream[29]	PVBusSlave
CMN650R2.cmn600_cache.upstream[2]	PVBusSlave
CMN650R2.cmn600_cache.upstream[30]	PVBusSlave
CMN650R2.cmn600_cache.upstream[31]	PVBusSlave
CMN650R2.cmn600_cache.upstream[32]	PVBusSlave
CMN650R2.cmn600_cache.upstream[33]	PVBusSlave
CMN650R2.cmn600_cache.upstream[3]	PVBusSlave
CMN650R2.cmn600_cache.upstream[4]	PVBusSlave
CMN650R2.cmn600_cache.upstream[5]	PVBusSlave
CMN650R2.cmn600_cache.upstream[6]	PVBusSlave
CMN650R2.cmn600_cache.upstream[7]	PVBusSlave
CMN650R2.cmn600_cache.upstream[8]	PVBusSlave
CMN650R2.cmn600_cache.upstream[9]	PVBusSlave
CMN650R2.hnf_exclusive_monitor	PVBusExclusiveMonitor
CMN650R2.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CMN650R2.hni_exclusive_monitor0	PVBusExclusiveMonitor
CMN650R2.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CMN650R2.hni_exclusive_monitor1	PVBusExclusiveMonitor
CMN650R2.hni_exclusive_monitor1.bus_mapper	PVBusMapper
CMN650R2.hni_exclusive_monitor2	PVBusExclusiveMonitor
CMN650R2.hni_exclusive_monitor2.bus_mapper	PVBusMapper
CMN650R2.ocm_decoder	PVBusMapper
CMN650R2.ocm_exclusive_monitor	PVBusExclusiveMonitor

InstanceName	ComponentName
CMN650R2.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CMN650R2.snf_mapper	PVBusMapper

CMN650R2 contains the following CADI targets:

- CMN650R2

About CMN650R2

- For information about the changes in this release, see the Fast Models Portfolio release notes.
- Major IP revisions (rX) are modeled and are controlled by the `revision` parameter.
- Minor IP revisions (pY) are not modeled. There are no functional differences in the model between pY revisions, with the exception of the `periph_id2` register.
- To configure the model, you must have installed Arm® Socrates™. The `mesh_config_file` parameter defines the mesh placement of the CHI nodes. Set it to the name of the yml configuration file emitted by Socrates. You must use version r1p4-01rel0 or later of Socrates to generate the file. For more information about Socrates, see [Arm Socrates](#) on Arm Developer. To download it, contact [Arm Technical Support](#).
- Interconnect models are based on the TRM description and do not typically model RTL defects.

The model supports the following features:

- rnf, rni/rnd, hni, and snf/sbsx interface ports. The mapping between the port number and `NodeId` is based on the `NodeId` index. For example, RNF2 controls `pvbuss_s_rnf[2]`. Its index is specified in the `node_info` register as `logical_id`. Similar behavior can be expected for HN-I.

If both RN-D and RN-I nodes are present, then all starting rni ports are mapped to RN-D nodes and then the RN-I nodes. For example, for CMN650 with two RN-D nodes, one RN-I node, and given each RN-I or RN-D node controls three interface ports, `pvbuss_s_rni[0-2]` maps to RND0, `pvbuss_s_rni[3-5]` maps to RND1 and `pvbuss_s_rni[6-8]` maps to RNI0.

Similarly, SN-F and SBSX nodes are mapped to `pvbuss_m_snf[]` ports where starting ports are mapped to SN-F and then SBSX nodes.

- Mapping `HN*_SLC_SIZE_PARAM` values to cache sizes:
 - -8: OKB where `HN*_SLC_NUM_WAYS_PARAM=16`.



This value is not supported by the model.

- -2: 128KB where `HN*_SLC_NUM_WAYS_PARAM=16`
- -1: 256KB where `HN*_SLC_NUM_WAYS_PARAM=16`
- 0: 512KB where `HN*_SLC_NUM_WAYS_PARAM=16`
- 1: 1MB where `HN*_SLC_NUM_WAYS_PARAM=16`

- 2: 2MB where `HN*_SLC_NUM_WAYS_PARAM=16`
- 3: 3MB where `HN*_SLC_NUM_WAYS_PARAM=12`
- 3: 4MB where `HN*_SLC_NUM_WAYS_PARAM=16`
- The Address Based Flush (ABF) feature is available. After it has started, the ABF completes instantly. The ABF implementation has the following limitations:
 - In the Fast Model, any HN-F in the interconnect can perform an ABF on any address range even if that range is not part of the System Cache Group (SCG) that the HN-F belongs to.
 - Clearing the OCM with an ABF is not supported. A trace has been added to highlight this.
 - When `HNF_ABF_PR.abf_mode` is Reserved and `HNF_ABF_PR.abf_enable` is set, an error is printed and the ABF status register is set to complete successfully, but no operation takes place.
 - Because an ABF completes instantly, the `SF_PM_TRANSITION_ABORT` state cannot be reached. Contact [Arm Technical Support](#) with questions about support for this error condition.
 - The optional interrupt `INTREQPPU` is not supported.

Model limitations

- Out of scope:
 - PMU counters are not supported. Counter registers are implemented as RAZ.
 - QoS is not supported and all related registers are RAZ/WI.
 - Error injection and error generation are not supported. All error registers are RAZ/WI.
 - Power, clock, and interrupt signals are not supported.

The P-channel (power) signals `PREQ_LOGIC`, `PSTATE_LOGIC`, `PACCEPT_LOGIC`, `PDENY_LOGIC`, and `PACTIVE_LOGIC`, are not implemented. The model behaves as if `PSTATE_LOGIC` is 5'b00000. So the initial power state of the HN-F nodes is NOSFSLC/OFF. This is reflected in the reset values of `cmn_hns_ppu_pwpr`, which was named `por_hnf_ppu_pwpr` in earlier IP revisions. Software must write the `*_ppu_pwpr` registers to change the power state. Both HAM and FAM power states enable all of the SLC ways.

- Protocol credits and flit buffers are not supported.
- Snoop filtering is not supported.
- Prefetch Target operations are not supported.
- The MTE feature has been minimally tested and the functionality cannot be guaranteed. MTE error injection and detection are not supported.
- An access to a reserved or nonexistent register does not abort. It returns `pv::Tx_Data::TX_OK` and is RAZ/WI.
- Topologies with three I/O masters connected through an RN-I bridge device have been tested. Configurations where multiple I/O masters share a single ACE-lite port have not been tested.
- RNSAMs external to the mesh network are not functionally supported.



Note

Model parameter `force_rnsam_internal` is true by default. If this parameter is set to false and the topology has an RNFx (non-ESAM), the external bit (bit[31]) of the RNSAM children pointer is enabled. This bit is the only functionality governed by this parameter.

- By default, HN-F hashing uses the address [MAX:12] instead of the actual address [MAX:6], due to the DMI mechanism in the model. Enable the parameter `enable_rnsam_to_hnf_wider_hash` to make hashing logic use the actual address [MAX:6], but this might reduce the simulation speed of the model.
- OKB SLC data RAM and tag RAM are not supported.
- Source-based SLC cache partitioning is not supported.
- Way-based SLC cache partitioning is not supported.
- SLC/LCC partitioning dynamic allocation is not supported.
- Remote chip addresses incorrectly allocate a line in the SLC when `cache_state_modelled=true`. This can cause unexpected cache behavior that does not match the RTL, exhausting the SLC earlier or extra evictions.
- The fields in the `mvp_device_port_connect_info` register corresponding to the number of device credit slices and the number of CAL credit slices are not supported.
- SN-F hashing in HNSAM is not supported. HNSAM SN-F hashing is a single memory range mapped to multiple SN-Fs. Example HNFSAM_Mem_Map:

```
0x00000001000 - 0x00000002000 SNF0 SNF1
```

The model supports mapping HN-F to a single SN-F. Example HNFSAM_Mem_Map:

```
0x00000001000 - 0x00000002000 SNF0
0x00000002000 - 0x00000003000 SNF1
```

- HN-Fs with different SLC sizes in the same configuration are not supported.
- Transactions to addresses unused by Device registers in the Configuration Register Space are always routed to the Configuration Space even when the SAM is configured to use those addresses.
- GIC communication over A4S ports is not supported.
- MPAM features are not supported in the model, and the configuration-dependent register reset values do not match the RTL. Some of the registers (`por_hnf_mpam_idr` to `por_hnf_mpam_mbwumon_idr`) shared between security modes are not present in MPAM_S.
- The following limitations apply to System Cache Groups and Hash Target Groups:
 - A mix of local and remote HN-F targets is not supported.
 - CCG/CXRH target IDs in the HN target ID table are not supported (`sys_cache_grp_hn_nodeid_reg`).
 - The `hashed_target_grp_hnp_nodeid_reg` which is used for CCG/HN-P target IDs is not supported.

- System/Hash Target Groups only support HN-Fs.
- AXID hashing across HN-P/CCGs is not supported.
- The MPAM reset values described in `por_hnf_mpam_ns_por_hnf_mpam_idr[31:24]` default to the not-supported state. The model parameter `hnf_mpam_idr_override` can be used to override this but does not implement any of the functionality.
- MPAM_S `secure_register_groups_override` is not supported.
- No specific Debug Trace (DT) node functionality is supported. See [4. Plug-ins for Fast Models](#) on page 4138 for information about using plug-ins for trace.
- For Power State Transition related flush/disable of the SLC due to HNF_PPU_PWPR register, only the Operational Mode field (bits[7:4]) is considered. The Power Policy field (bits[3:0]) is ignored.
- Setting HAM mode for the Operational Mode field (bits[7:4]) of the HNF_PPU_PWPR is equivalent to FAM mode. The SLCH2 is neither flushed nor disabled according to the Power Policy field (bits[3:0]).
- Only hashing granularities of 4096B and above are supported.
- The OCM does not keep track of the total memory used across the S/NS security world and does not perform any eviction either. It is up to the software to make sure it does not oversubscribe the OCM.
- All RNs are assumed to have the same HN mapping. Programming RNSAMs differently between RNs is not supported.
- RNSAM cannot be programmed with the same HN-F target in a hashed and non-hashed memory region.
- `RNSAM_STATUS.default_target_type` and `RNSAM_STATUS.default_nodeid` are used as the default target only when `RNSAM_STATUS.use_default_node=1`. Otherwise, when a txn is sent into the model that does not belong to any of the address regions programmed in the RNSAM, it is routed to the HN-D irrespective of what is programmed in the `RNSAM_STATUS.default_target_type` and `RNSAM_STATUS.default_nodeid` fields.
- Address bit masking is not supported in RNSAM or HN-F SAM.
- HN-F SAM limitations:
 - Address masking in default hash regions in HN-F SAM is not supported.
 - Hashing in default hash regions in HN-F SAM is not supported. However, trace `SNF_HASHING_Target_SNF` displays the target SNF as if 3SN/6SN hashing were enabled.



The transaction is always routed to the SNO nodeid programmed in the SAM_CONTROL register.

-
- Re-programming regions in HN-F SAM is not tested.
 - Hashing across CCGs in HN-F SAM is not supported.
 - CAL2 support for HN-P and RN-D is not tested.

- The maximum number of 256 RN-Fs is not verified. 74 is the largest number tested.
- The maximum number of 40 SNs is not verified. 20 is the largest number tested.
- The maximum number of 36 RN-Is is not verified. 16 is the largest number tested.
- The maximum number of 16 HN-Is is not verified. 5 is the largest number tested.
- Early DVM completion is not supported.
- CCIX port to port forwarding is not supported.
- No support for up to 512 CXRAs with no RAID aliasing and 256 RN-Fs on a single chip.
- Each HN-F can support tracking of up to 512 logical processors for exclusive operations. However, the value of the RO field `num_exc1` in the HN-F unit info register cannot exceed 255.

About the `debug_force_snoop` parameter

The CMN interconnect normally starts up with snooping disabled.

The parameters `rnf_sci_enable` and `rni_sci_enable` determine which connections are managed by the System Coherency Interface. Connections that are managed by the System Coherency Interface look after themselves for entering and leaving the coherency domain and `debug_force_snoop` has no effect on those ports.

However, for connections that are not managed by the System Coherency Interface, `debug_force_snoop` enables the system upstream of a port to be snooped if the upstream is not in reset and if `acchannelen_rnf` allows it.

Therefore, software does not need to know how to turn snoops in the interconnect on and off. The reset state of the upstream rnf ports is determined by the `rnf_upstream_reset_state[]` signals and must be connected for any port that could go into reset and you want managed by `debug_force_snoop`.

If the connection is managed by SCI or can never go into reset then you need not connect this and the interconnect assumes that the upstream system is not in reset and will always enable the snoops if `ACCHANNELENSx` allows it.

If you fail to connect `rnf_upstream_reset_state[]` correctly, then the upstream system might receive snoop messages while in reset and it will complain that it received a snoop request while it was in reset.

If software overrides the enables then the next change to `rnf_upstream_reset_state[]` or the interconnect reset will overwrite the software values.

Using this operation along with software controlling the enables could confuse the software. This option does not remove the responsibility of the upstream system to clean any shared dirty data from its caches before going into reset.



This parameter is for debug purposes only. Do not use it instead of correctly configuring CMN.

Ports for CMN650R2

Table 3-891: Ports

Name	Protocol	Type	Description
event_clusters	Signal	Peer	CPU event communication signal from the clusters.
event_downstream_link_signal[10]	Signal	Master	CPU event communication signal to the CMLHub. Event from the CMN towards the Hub NOTE that these are virtual "links" on underlying PCIe hardware bus.
event_upstream_link_signal[10]	Signal	Slave	Event from the Hub towards the CMN
pvbus_m_cml	PVBus	Master	CML downstream ports
pvbus_m_cml_cfg	PVBus	Master	CML downstream hub configuration port
pvbus_m_hni[16]	PVBus	Master	HNI downstream ports.
pvbus_m_snf[40]	PVBus	Master	SNF downstream port.
pvbus_s_apb	PVBus	Slave	APB interface port.
pvbus_s_cml	PVBus	Slave	CML upstream ports
pvbus_s_rnf[256]	PVBus	Slave	RNF upstream ports.
pvbus_s_rni[108]	PVBus	Slave	RNI upstream ports.
reset_in	Signal	Slave	Reset signal.
rnf_sci_s[256]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-F in and out of the coherency domain.
rnf_upstream_reset_in[256]	Signal	Slave	Used by the interconnect to determine the reset state of the RNF manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.
rni_sci_s[108]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-D in and out of the coherency domain.
rni_upstream_reset_in[108]	Signal	Slave	Used by the interconnect to determine the reset state of the RNI manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.

Parameters for CMN650R2

acchannelen_rnf

DEPRECATED: Will be removed after FM 11.18. Use 'rnf_sci_enable' instead. For each rnf port, indicates if the port is populated with a snoop responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

acchannelen_rni

DEPRECATED: Will be removed after FM 11.18. Use 'rni_sci_enable' instead. For each rni port, indicates if the port is populated with a dvm responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x0

debug_force_snoop

The CMN650R2 interconnect will normally start with snooping disabled. The parameter rnf_sci_enable and rni_sci_enable determines which connections are managed by the System Coherency Interface. For connections that are managed by the System Coherency Interface, then they will look after themselves for entering and leaving the coherency domain and this parameter has no effect for those ports. However, for connections that are _not_ managed by the System Coherency Interface, this parameter enables the upstream system of a port to be snooped if the upstream is not in reset and if acchannelen_rnf allows it. NOTE: This parameter is for debug purpose only Do not use this parameter instead of correctly configuring CMN. Refer CMN650R2 documentation page for more info about this parameter in fast models reference manual

Type

bool

Default value

0x0

disable_CML_port

If true the model won't connect the CML port when there are nodes which support the CML feature (ie CXG nodes).

Type

bool

Default value

0x0

enable_logger

Enable PVBUSLoggers for the downstream ports in the CMN model.

Type

bool

Default value

0x0

enable_rnsam_to_hnf_wider_hash

Enable support of wider hash for the RNSAM to HNF communication. If this variable enabled, then bits[47:6] from PA used in hashing function. By default it is [47:12].

Type

bool

Default value

0x0

force_rnsam_internal

Force all RNSAMs to be internal independently of the mesh topology.

Type

bool

Default value

0x1

hnf_mpam_idr_override

Set to override hnf_mpam_idr[31:24] value. Bit[28] is Reserved and is ignored.

Type

int

Default value

0x0

mesh_config_file

Name of a file containing mesh placement of CMN650R2 components.

Type

string

Default value

/tmp/plgbuild/abs_build/1378878_61223/trunk/work/fastsim/autodocs/.../Platforms/
LISA/CSS/Kits4/cmn650_rdv1.yml

periphbase

Value for PERIPHBASE. Bits [27:0] are treated 0

Type

int

Default value

0x20000000

print_cmn_ccix_config

Print information about the CCIX configuration.

Type

bool

Default value

0x0

print_cmn_config

Print the mesh topology and children pointers acquired from the YAML file.

Type

bool

Default value

0x0

revision

Component revision. Currently supports r2p0.

Type

string

Default value

r2p0

rnf_sci_enable

For each rnf port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

rni_sci_enable

For each rni port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

show_banner

Show component banner: 0=>supress entire banner, 1=>suppress config file, 2+=>show full banner.

Type

int

Default value

0x2

skip_cmn_config_check

Skip any topology configuration checks. The maximum number of devices per type not verified.

Type

bool

Default value

0x0

use_yaml_periphbase

Use yaml param CFGM_PERIPHBASE_PARAM to specify periphbase address. If false, model parameter 'periphbase' will be used.

Type

bool

Default value

0x0

3.10.18 CMN700

CMN700 Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-892: IP revisions support

Revision	Quality level
r0p0	Full support
r1p0	Full support
r2p0	Full support
r3p0	Full support
r3p3	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- a4s_logicalid

Iris and MTI instances for CMN700

This model has the following Iris instances:

Table 3-893: CMN700 Iris instances

InstanceName	ComponentName
CMN700	CMN700
CMN700.bus_slave_ocm_NS	PVBusSlave
CMN700.bus_slave_ocm_S	PVBusSlave
CMN700.cmn600_cache	PVCache
CMN700.cmn600_cache.upstream[0]	PVBusSlave
CMN700.cmn600_cache.upstream[10]	PVBusSlave
CMN700.cmn600_cache.upstream[11]	PVBusSlave
CMN700.cmn600_cache.upstream[12]	PVBusSlave
CMN700.cmn600_cache.upstream[13]	PVBusSlave
CMN700.cmn600_cache.upstream[14]	PVBusSlave
CMN700.cmn600_cache.upstream[15]	PVBusSlave
CMN700.cmn600_cache.upstream[16]	PVBusSlave
CMN700.cmn600_cache.upstream[17]	PVBusSlave
CMN700.cmn600_cache.upstream[18]	PVBusSlave
CMN700.cmn600_cache.upstream[19]	PVBusSlave
CMN700.cmn600_cache.upstream[1]	PVBusSlave
CMN700.cmn600_cache.upstream[20]	PVBusSlave
CMN700.cmn600_cache.upstream[21]	PVBusSlave
CMN700.cmn600_cache.upstream[22]	PVBusSlave
CMN700.cmn600_cache.upstream[23]	PVBusSlave
CMN700.cmn600_cache.upstream[24]	PVBusSlave
CMN700.cmn600_cache.upstream[25]	PVBusSlave
CMN700.cmn600_cache.upstream[26]	PVBusSlave
CMN700.cmn600_cache.upstream[27]	PVBusSlave
CMN700.cmn600_cache.upstream[28]	PVBusSlave
CMN700.cmn600_cache.upstream[29]	PVBusSlave
CMN700.cmn600_cache.upstream[2]	PVBusSlave
CMN700.cmn600_cache.upstream[30]	PVBusSlave
CMN700.cmn600_cache.upstream[31]	PVBusSlave
CMN700.cmn600_cache.upstream[32]	PVBusSlave
CMN700.cmn600_cache.upstream[33]	PVBusSlave
CMN700.cmn600_cache.upstream[34]	PVBusSlave
CMN700.cmn600_cache.upstream[35]	PVBusSlave
CMN700.cmn600_cache.upstream[36]	PVBusSlave
CMN700.cmn600_cache.upstream[37]	PVBusSlave

InstanceName	ComponentName
CMN700.cmn600_cache.upstream[3]	PVBusSlave
CMN700.cmn600_cache.upstream[4]	PVBusSlave
CMN700.cmn600_cache.upstream[5]	PVBusSlave
CMN700.cmn600_cache.upstream[6]	PVBusSlave
CMN700.cmn600_cache.upstream[7]	PVBusSlave
CMN700.cmn600_cache.upstream[8]	PVBusSlave
CMN700.cmn600_cache.upstream[9]	PVBusSlave
CMN700.cmn700_tag_cache	CMN_TAG_CACHE
CMN700.cmn700_tag_cache.metadata_controller0	ExportTest.CMN700.cmn700_tag_cachemetadata_controller0
CMN700.cmn700_tag_cache. metadata_controller0.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller1	ExportTest.CMN700.cmn700_tag_cachemetadata_controller1
CMN700.cmn700_tag_cache. metadata_controller1.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller10	ExportTest.CMN700.cmn700_tag_cachemetadata_controller10
CMN700.cmn700_tag_cache. metadata_controller10.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller100	ExportTest.CMN700.cmn700_tag_cachemetadata_controller100
CMN700.cmn700_tag_cache. metadata_controller100.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller101	ExportTest.CMN700.cmn700_tag_cachemetadata_controller101
CMN700.cmn700_tag_cache. metadata_controller101.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller102	ExportTest.CMN700.cmn700_tag_cachemetadata_controller102
CMN700.cmn700_tag_cache. metadata_controller102.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller103	ExportTest.CMN700.cmn700_tag_cachemetadata_controller103
CMN700.cmn700_tag_cache. metadata_controller103.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller104	ExportTest.CMN700.cmn700_tag_cachemetadata_controller104
CMN700.cmn700_tag_cache. metadata_controller104.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller105	ExportTest.CMN700.cmn700_tag_cachemetadata_controller105
CMN700.cmn700_tag_cache. metadata_controller105.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller106	ExportTest.CMN700.cmn700_tag_cachemetadata_controller106
CMN700.cmn700_tag_cache. metadata_controller106.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller107	ExportTest.CMN700.cmn700_tag_cachemetadata_controller107
CMN700.cmn700_tag_cache. metadata_controller107.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller108	ExportTest.CMN700.cmn700_tag_cachemetadata_controller108

InstanceName	ComponentName
CMN700.cmn700_tag_cache. metadata_controller108.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller109	ExportTest.CMN700.cmn700_tag_cachemetadata_controller109
CMN700.cmn700_tag_cache. metadata_controller109.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller11	ExportTest.CMN700.cmn700_tag_cachemetadata_controller11
CMN700.cmn700_tag_cache. metadata_controller11.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller110	ExportTest.CMN700.cmn700_tag_cachemetadata_controller110
CMN700.cmn700_tag_cache. metadata_controller110.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller111	ExportTest.CMN700.cmn700_tag_cachemetadata_controller111
CMN700.cmn700_tag_cache. metadata_controller111.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller112	ExportTest.CMN700.cmn700_tag_cachemetadata_controller112
CMN700.cmn700_tag_cache. metadata_controller112.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller113	ExportTest.CMN700.cmn700_tag_cachemetadata_controller113
CMN700.cmn700_tag_cache. metadata_controller113.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller114	ExportTest.CMN700.cmn700_tag_cachemetadata_controller114
CMN700.cmn700_tag_cache. metadata_controller114.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller115	ExportTest.CMN700.cmn700_tag_cachemetadata_controller115
CMN700.cmn700_tag_cache. metadata_controller115.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller116	ExportTest.CMN700.cmn700_tag_cachemetadata_controller116
CMN700.cmn700_tag_cache. metadata_controller116.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller117	ExportTest.CMN700.cmn700_tag_cachemetadata_controller117
CMN700.cmn700_tag_cache. metadata_controller117.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller118	ExportTest.CMN700.cmn700_tag_cachemetadata_controller118
CMN700.cmn700_tag_cache. metadata_controller118.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller119	ExportTest.CMN700.cmn700_tag_cachemetadata_controller119
CMN700.cmn700_tag_cache. metadata_controller119.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller12	ExportTest.CMN700.cmn700_tag_cachemetadata_controller12
CMN700.cmn700_tag_cache. metadata_controller12.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller120	ExportTest.CMN700.cmn700_tag_cachemetadata_controller120
CMN700.cmn700_tag_cache. metadata_controller120.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.metadata_controller121	ExportTest.CMN700.cmn700_tag_cachemetadata_controller121
CMN700.cmn700_tag_cache. metadata_controller121.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller122	ExportTest.CMN700.cmn700_tag_cachemetadata_controller122
CMN700.cmn700_tag_cache. metadata_controller122.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller123	ExportTest.CMN700.cmn700_tag_cachemetadata_controller123
CMN700.cmn700_tag_cache. metadata_controller123.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller124	ExportTest.CMN700.cmn700_tag_cachemetadata_controller124
CMN700.cmn700_tag_cache. metadata_controller124.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller125	ExportTest.CMN700.cmn700_tag_cachemetadata_controller125
CMN700.cmn700_tag_cache. metadata_controller125.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller126	ExportTest.CMN700.cmn700_tag_cachemetadata_controller126
CMN700.cmn700_tag_cache. metadata_controller126.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller127	ExportTest.CMN700.cmn700_tag_cachemetadata_controller127
CMN700.cmn700_tag_cache. metadata_controller127.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller13	ExportTest.CMN700.cmn700_tag_cachemetadata_controller13
CMN700.cmn700_tag_cache. metadata_controller13.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller14	ExportTest.CMN700.cmn700_tag_cachemetadata_controller14
CMN700.cmn700_tag_cache. metadata_controller14.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller15	ExportTest.CMN700.cmn700_tag_cachemetadata_controller15
CMN700.cmn700_tag_cache. metadata_controller15.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller16	ExportTest.CMN700.cmn700_tag_cachemetadata_controller16
CMN700.cmn700_tag_cache. metadata_controller16.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller17	ExportTest.CMN700.cmn700_tag_cachemetadata_controller17
CMN700.cmn700_tag_cache. metadata_controller17.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller18	ExportTest.CMN700.cmn700_tag_cachemetadata_controller18
CMN700.cmn700_tag_cache. metadata_controller18.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller19	ExportTest.CMN700.cmn700_tag_cachemetadata_controller19
CMN700.cmn700_tag_cache. metadata_controller19.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller2	ExportTest.CMN700.cmn700_tag_cachemetadata_controller2

InstanceName	ComponentName
CMN700.cmn700_tag_cache. metadata_controller2.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller20	ExportTest.CMN700.cmn700_tag_cachemetadata_controller20
CMN700.cmn700_tag_cache. metadata_controller20.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller21	ExportTest.CMN700.cmn700_tag_cachemetadata_controller21
CMN700.cmn700_tag_cache. metadata_controller21.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller22	ExportTest.CMN700.cmn700_tag_cachemetadata_controller22
CMN700.cmn700_tag_cache. metadata_controller22.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller23	ExportTest.CMN700.cmn700_tag_cachemetadata_controller23
CMN700.cmn700_tag_cache. metadata_controller23.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller24	ExportTest.CMN700.cmn700_tag_cachemetadata_controller24
CMN700.cmn700_tag_cache. metadata_controller24.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller25	ExportTest.CMN700.cmn700_tag_cachemetadata_controller25
CMN700.cmn700_tag_cache. metadata_controller25.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller26	ExportTest.CMN700.cmn700_tag_cachemetadata_controller26
CMN700.cmn700_tag_cache. metadata_controller26.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller27	ExportTest.CMN700.cmn700_tag_cachemetadata_controller27
CMN700.cmn700_tag_cache. metadata_controller27.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller28	ExportTest.CMN700.cmn700_tag_cachemetadata_controller28
CMN700.cmn700_tag_cache. metadata_controller28.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller29	ExportTest.CMN700.cmn700_tag_cachemetadata_controller29
CMN700.cmn700_tag_cache. metadata_controller29.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller3	ExportTest.CMN700.cmn700_tag_cachemetadata_controller3
CMN700.cmn700_tag_cache. metadata_controller3.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller30	ExportTest.CMN700.cmn700_tag_cachemetadata_controller30
CMN700.cmn700_tag_cache. metadata_controller30.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller31	ExportTest.CMN700.cmn700_tag_cachemetadata_controller31
CMN700.cmn700_tag_cache. metadata_controller31.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller32	ExportTest.CMN700.cmn700_tag_cachemetadata_controller32
CMN700.cmn700_tag_cache. metadata_controller32.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.metadata_controller33	ExportTest.CMN700.cmn700_tag_cachemetadata_controller33
CMN700.cmn700_tag_cache. metadata_controller33.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller34	ExportTest.CMN700.cmn700_tag_cachemetadata_controller34
CMN700.cmn700_tag_cache. metadata_controller34.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller35	ExportTest.CMN700.cmn700_tag_cachemetadata_controller35
CMN700.cmn700_tag_cache. metadata_controller35.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller36	ExportTest.CMN700.cmn700_tag_cachemetadata_controller36
CMN700.cmn700_tag_cache. metadata_controller36.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller37	ExportTest.CMN700.cmn700_tag_cachemetadata_controller37
CMN700.cmn700_tag_cache. metadata_controller37.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller38	ExportTest.CMN700.cmn700_tag_cachemetadata_controller38
CMN700.cmn700_tag_cache. metadata_controller38.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller39	ExportTest.CMN700.cmn700_tag_cachemetadata_controller39
CMN700.cmn700_tag_cache. metadata_controller39.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller4	ExportTest.CMN700.cmn700_tag_cachemetadata_controller4
CMN700.cmn700_tag_cache. metadata_controller4.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller40	ExportTest.CMN700.cmn700_tag_cachemetadata_controller40
CMN700.cmn700_tag_cache. metadata_controller40.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller41	ExportTest.CMN700.cmn700_tag_cachemetadata_controller41
CMN700.cmn700_tag_cache. metadata_controller41.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller42	ExportTest.CMN700.cmn700_tag_cachemetadata_controller42
CMN700.cmn700_tag_cache. metadata_controller42.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller43	ExportTest.CMN700.cmn700_tag_cachemetadata_controller43
CMN700.cmn700_tag_cache. metadata_controller43.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller44	ExportTest.CMN700.cmn700_tag_cachemetadata_controller44
CMN700.cmn700_tag_cache. metadata_controller44.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller45	ExportTest.CMN700.cmn700_tag_cachemetadata_controller45
CMN700.cmn700_tag_cache. metadata_controller45.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller46	ExportTest.CMN700.cmn700_tag_cachemetadata_controller46

InstanceName	ComponentName
CMN700.cmn700_tag_cache. metadata_controller46.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller47	ExportTest.CMN700.cmn700_tag_cachemetadata_controller47
CMN700.cmn700_tag_cache. metadata_controller47.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller48	ExportTest.CMN700.cmn700_tag_cachemetadata_controller48
CMN700.cmn700_tag_cache. metadata_controller48.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller49	ExportTest.CMN700.cmn700_tag_cachemetadata_controller49
CMN700.cmn700_tag_cache. metadata_controller49.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller5	ExportTest.CMN700.cmn700_tag_cachemetadata_controller5
CMN700.cmn700_tag_cache. metadata_controller5.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller50	ExportTest.CMN700.cmn700_tag_cachemetadata_controller50
CMN700.cmn700_tag_cache. metadata_controller50.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller51	ExportTest.CMN700.cmn700_tag_cachemetadata_controller51
CMN700.cmn700_tag_cache. metadata_controller51.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller52	ExportTest.CMN700.cmn700_tag_cachemetadata_controller52
CMN700.cmn700_tag_cache. metadata_controller52.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller53	ExportTest.CMN700.cmn700_tag_cachemetadata_controller53
CMN700.cmn700_tag_cache. metadata_controller53.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller54	ExportTest.CMN700.cmn700_tag_cachemetadata_controller54
CMN700.cmn700_tag_cache. metadata_controller54.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller55	ExportTest.CMN700.cmn700_tag_cachemetadata_controller55
CMN700.cmn700_tag_cache. metadata_controller55.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller56	ExportTest.CMN700.cmn700_tag_cachemetadata_controller56
CMN700.cmn700_tag_cache. metadata_controller56.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller57	ExportTest.CMN700.cmn700_tag_cachemetadata_controller57
CMN700.cmn700_tag_cache. metadata_controller57.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller58	ExportTest.CMN700.cmn700_tag_cachemetadata_controller58
CMN700.cmn700_tag_cache. metadata_controller58.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller59	ExportTest.CMN700.cmn700_tag_cachemetadata_controller59
CMN700.cmn700_tag_cache. metadata_controller59.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.metadata_controller6	ExportTest.CMN700.cmn700_tag_cachemetadata_controller6
CMN700.cmn700_tag_cache. metadata_controller6.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller60	ExportTest.CMN700.cmn700_tag_cachemetadata_controller60
CMN700.cmn700_tag_cache. metadata_controller60.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller61	ExportTest.CMN700.cmn700_tag_cachemetadata_controller61
CMN700.cmn700_tag_cache. metadata_controller61.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller62	ExportTest.CMN700.cmn700_tag_cachemetadata_controller62
CMN700.cmn700_tag_cache. metadata_controller62.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller63	ExportTest.CMN700.cmn700_tag_cachemetadata_controller63
CMN700.cmn700_tag_cache. metadata_controller63.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller64	ExportTest.CMN700.cmn700_tag_cachemetadata_controller64
CMN700.cmn700_tag_cache. metadata_controller64.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller65	ExportTest.CMN700.cmn700_tag_cachemetadata_controller65
CMN700.cmn700_tag_cache. metadata_controller65.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller66	ExportTest.CMN700.cmn700_tag_cachemetadata_controller66
CMN700.cmn700_tag_cache. metadata_controller66.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller67	ExportTest.CMN700.cmn700_tag_cachemetadata_controller67
CMN700.cmn700_tag_cache. metadata_controller67.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller68	ExportTest.CMN700.cmn700_tag_cachemetadata_controller68
CMN700.cmn700_tag_cache. metadata_controller68.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller69	ExportTest.CMN700.cmn700_tag_cachemetadata_controller69
CMN700.cmn700_tag_cache. metadata_controller69.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller7	ExportTest.CMN700.cmn700_tag_cachemetadata_controller7
CMN700.cmn700_tag_cache. metadata_controller7.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller70	ExportTest.CMN700.cmn700_tag_cachemetadata_controller70
CMN700.cmn700_tag_cache. metadata_controller70.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller71	ExportTest.CMN700.cmn700_tag_cachemetadata_controller71
CMN700.cmn700_tag_cache. metadata_controller71.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller72	ExportTest.CMN700.cmn700_tag_cachemetadata_controller72

InstanceName	ComponentName
CMN700.cmn700_tag_cache. metadata_controller72.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller73	ExportTest.CMN700.cmn700_tag_cachemetadata_controller73
CMN700.cmn700_tag_cache. metadata_controller73.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller74	ExportTest.CMN700.cmn700_tag_cachemetadata_controller74
CMN700.cmn700_tag_cache. metadata_controller74.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller75	ExportTest.CMN700.cmn700_tag_cachemetadata_controller75
CMN700.cmn700_tag_cache. metadata_controller75.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller76	ExportTest.CMN700.cmn700_tag_cachemetadata_controller76
CMN700.cmn700_tag_cache. metadata_controller76.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller77	ExportTest.CMN700.cmn700_tag_cachemetadata_controller77
CMN700.cmn700_tag_cache. metadata_controller77.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller78	ExportTest.CMN700.cmn700_tag_cachemetadata_controller78
CMN700.cmn700_tag_cache. metadata_controller78.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller79	ExportTest.CMN700.cmn700_tag_cachemetadata_controller79
CMN700.cmn700_tag_cache. metadata_controller79.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller8	ExportTest.CMN700.cmn700_tag_cachemetadata_controller8
CMN700.cmn700_tag_cache. metadata_controller8.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller80	ExportTest.CMN700.cmn700_tag_cachemetadata_controller80
CMN700.cmn700_tag_cache. metadata_controller80.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller81	ExportTest.CMN700.cmn700_tag_cachemetadata_controller81
CMN700.cmn700_tag_cache. metadata_controller81.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller82	ExportTest.CMN700.cmn700_tag_cachemetadata_controller82
CMN700.cmn700_tag_cache. metadata_controller82.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller83	ExportTest.CMN700.cmn700_tag_cachemetadata_controller83
CMN700.cmn700_tag_cache. metadata_controller83.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller84	ExportTest.CMN700.cmn700_tag_cachemetadata_controller84
CMN700.cmn700_tag_cache. metadata_controller84.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller85	ExportTest.CMN700.cmn700_tag_cachemetadata_controller85
CMN700.cmn700_tag_cache. metadata_controller85.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.metadata_controller86	ExportTest.CMN700.cmn700_tag_cachemetadata_controller86
CMN700.cmn700_tag_cache. metadata_controller86.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller87	ExportTest.CMN700.cmn700_tag_cachemetadata_controller87
CMN700.cmn700_tag_cache. metadata_controller87.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller88	ExportTest.CMN700.cmn700_tag_cachemetadata_controller88
CMN700.cmn700_tag_cache. metadata_controller88.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller89	ExportTest.CMN700.cmn700_tag_cachemetadata_controller89
CMN700.cmn700_tag_cache. metadata_controller89.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller9	ExportTest.CMN700.cmn700_tag_cachemetadata_controller9
CMN700.cmn700_tag_cache. metadata_controller9.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller90	ExportTest.CMN700.cmn700_tag_cachemetadata_controller90
CMN700.cmn700_tag_cache. metadata_controller90.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller91	ExportTest.CMN700.cmn700_tag_cachemetadata_controller91
CMN700.cmn700_tag_cache. metadata_controller91.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller92	ExportTest.CMN700.cmn700_tag_cachemetadata_controller92
CMN700.cmn700_tag_cache. metadata_controller92.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller93	ExportTest.CMN700.cmn700_tag_cachemetadata_controller93
CMN700.cmn700_tag_cache. metadata_controller93.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller94	ExportTest.CMN700.cmn700_tag_cachemetadata_controller94
CMN700.cmn700_tag_cache. metadata_controller94.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller95	ExportTest.CMN700.cmn700_tag_cachemetadata_controller95
CMN700.cmn700_tag_cache. metadata_controller95.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller96	ExportTest.CMN700.cmn700_tag_cachemetadata_controller96
CMN700.cmn700_tag_cache. metadata_controller96.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller97	ExportTest.CMN700.cmn700_tag_cachemetadata_controller97
CMN700.cmn700_tag_cache. metadata_controller97.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller98	ExportTest.CMN700.cmn700_tag_cachemetadata_controller98
CMN700.cmn700_tag_cache. metadata_controller98.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller99	ExportTest.CMN700.cmn700_tag_cachemetadata_controller99

InstanceName	ComponentName
CMN700.cmn700_tag_cache.metadata_controller99.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.remapper0	PVBusMapper
CMN700.cmn700_tag_cache.remapper1	PVBusMapper
CMN700.cmn700_tag_cache.remapper10	PVBusMapper
CMN700.cmn700_tag_cache.remapper100	PVBusMapper
CMN700.cmn700_tag_cache.remapper101	PVBusMapper
CMN700.cmn700_tag_cache.remapper102	PVBusMapper
CMN700.cmn700_tag_cache.remapper103	PVBusMapper
CMN700.cmn700_tag_cache.remapper104	PVBusMapper
CMN700.cmn700_tag_cache.remapper105	PVBusMapper
CMN700.cmn700_tag_cache.remapper106	PVBusMapper
CMN700.cmn700_tag_cache.remapper107	PVBusMapper
CMN700.cmn700_tag_cache.remapper108	PVBusMapper
CMN700.cmn700_tag_cache.remapper109	PVBusMapper
CMN700.cmn700_tag_cache.remapper11	PVBusMapper
CMN700.cmn700_tag_cache.remapper110	PVBusMapper
CMN700.cmn700_tag_cache.remapper111	PVBusMapper
CMN700.cmn700_tag_cache.remapper112	PVBusMapper
CMN700.cmn700_tag_cache.remapper113	PVBusMapper
CMN700.cmn700_tag_cache.remapper114	PVBusMapper
CMN700.cmn700_tag_cache.remapper115	PVBusMapper
CMN700.cmn700_tag_cache.remapper116	PVBusMapper
CMN700.cmn700_tag_cache.remapper117	PVBusMapper
CMN700.cmn700_tag_cache.remapper118	PVBusMapper
CMN700.cmn700_tag_cache.remapper119	PVBusMapper
CMN700.cmn700_tag_cache.remapper12	PVBusMapper
CMN700.cmn700_tag_cache.remapper120	PVBusMapper
CMN700.cmn700_tag_cache.remapper121	PVBusMapper
CMN700.cmn700_tag_cache.remapper122	PVBusMapper
CMN700.cmn700_tag_cache.remapper123	PVBusMapper
CMN700.cmn700_tag_cache.remapper124	PVBusMapper
CMN700.cmn700_tag_cache.remapper125	PVBusMapper
CMN700.cmn700_tag_cache.remapper126	PVBusMapper
CMN700.cmn700_tag_cache.remapper127	PVBusMapper
CMN700.cmn700_tag_cache.remapper13	PVBusMapper
CMN700.cmn700_tag_cache.remapper14	PVBusMapper
CMN700.cmn700_tag_cache.remapper15	PVBusMapper
CMN700.cmn700_tag_cache.remapper16	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.remapper17	PVBusMapper
CMN700.cmn700_tag_cache.remapper18	PVBusMapper
CMN700.cmn700_tag_cache.remapper19	PVBusMapper
CMN700.cmn700_tag_cache.remapper2	PVBusMapper
CMN700.cmn700_tag_cache.remapper20	PVBusMapper
CMN700.cmn700_tag_cache.remapper21	PVBusMapper
CMN700.cmn700_tag_cache.remapper22	PVBusMapper
CMN700.cmn700_tag_cache.remapper23	PVBusMapper
CMN700.cmn700_tag_cache.remapper24	PVBusMapper
CMN700.cmn700_tag_cache.remapper25	PVBusMapper
CMN700.cmn700_tag_cache.remapper26	PVBusMapper
CMN700.cmn700_tag_cache.remapper27	PVBusMapper
CMN700.cmn700_tag_cache.remapper28	PVBusMapper
CMN700.cmn700_tag_cache.remapper29	PVBusMapper
CMN700.cmn700_tag_cache.remapper3	PVBusMapper
CMN700.cmn700_tag_cache.remapper30	PVBusMapper
CMN700.cmn700_tag_cache.remapper31	PVBusMapper
CMN700.cmn700_tag_cache.remapper32	PVBusMapper
CMN700.cmn700_tag_cache.remapper33	PVBusMapper
CMN700.cmn700_tag_cache.remapper34	PVBusMapper
CMN700.cmn700_tag_cache.remapper35	PVBusMapper
CMN700.cmn700_tag_cache.remapper36	PVBusMapper
CMN700.cmn700_tag_cache.remapper37	PVBusMapper
CMN700.cmn700_tag_cache.remapper38	PVBusMapper
CMN700.cmn700_tag_cache.remapper39	PVBusMapper
CMN700.cmn700_tag_cache.remapper4	PVBusMapper
CMN700.cmn700_tag_cache.remapper40	PVBusMapper
CMN700.cmn700_tag_cache.remapper41	PVBusMapper
CMN700.cmn700_tag_cache.remapper42	PVBusMapper
CMN700.cmn700_tag_cache.remapper43	PVBusMapper
CMN700.cmn700_tag_cache.remapper44	PVBusMapper
CMN700.cmn700_tag_cache.remapper45	PVBusMapper
CMN700.cmn700_tag_cache.remapper46	PVBusMapper
CMN700.cmn700_tag_cache.remapper47	PVBusMapper
CMN700.cmn700_tag_cache.remapper48	PVBusMapper
CMN700.cmn700_tag_cache.remapper49	PVBusMapper
CMN700.cmn700_tag_cache.remapper5	PVBusMapper
CMN700.cmn700_tag_cache.remapper50	PVBusMapper
CMN700.cmn700_tag_cache.remapper51	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.remapper52	PVBusMapper
CMN700.cmn700_tag_cache.remapper53	PVBusMapper
CMN700.cmn700_tag_cache.remapper54	PVBusMapper
CMN700.cmn700_tag_cache.remapper55	PVBusMapper
CMN700.cmn700_tag_cache.remapper56	PVBusMapper
CMN700.cmn700_tag_cache.remapper57	PVBusMapper
CMN700.cmn700_tag_cache.remapper58	PVBusMapper
CMN700.cmn700_tag_cache.remapper59	PVBusMapper
CMN700.cmn700_tag_cache.remapper6	PVBusMapper
CMN700.cmn700_tag_cache.remapper60	PVBusMapper
CMN700.cmn700_tag_cache.remapper61	PVBusMapper
CMN700.cmn700_tag_cache.remapper62	PVBusMapper
CMN700.cmn700_tag_cache.remapper63	PVBusMapper
CMN700.cmn700_tag_cache.remapper64	PVBusMapper
CMN700.cmn700_tag_cache.remapper65	PVBusMapper
CMN700.cmn700_tag_cache.remapper66	PVBusMapper
CMN700.cmn700_tag_cache.remapper67	PVBusMapper
CMN700.cmn700_tag_cache.remapper68	PVBusMapper
CMN700.cmn700_tag_cache.remapper69	PVBusMapper
CMN700.cmn700_tag_cache.remapper7	PVBusMapper
CMN700.cmn700_tag_cache.remapper70	PVBusMapper
CMN700.cmn700_tag_cache.remapper71	PVBusMapper
CMN700.cmn700_tag_cache.remapper72	PVBusMapper
CMN700.cmn700_tag_cache.remapper73	PVBusMapper
CMN700.cmn700_tag_cache.remapper74	PVBusMapper
CMN700.cmn700_tag_cache.remapper75	PVBusMapper
CMN700.cmn700_tag_cache.remapper76	PVBusMapper
CMN700.cmn700_tag_cache.remapper77	PVBusMapper
CMN700.cmn700_tag_cache.remapper78	PVBusMapper
CMN700.cmn700_tag_cache.remapper79	PVBusMapper
CMN700.cmn700_tag_cache.remapper8	PVBusMapper
CMN700.cmn700_tag_cache.remapper80	PVBusMapper
CMN700.cmn700_tag_cache.remapper81	PVBusMapper
CMN700.cmn700_tag_cache.remapper82	PVBusMapper
CMN700.cmn700_tag_cache.remapper83	PVBusMapper
CMN700.cmn700_tag_cache.remapper84	PVBusMapper
CMN700.cmn700_tag_cache.remapper85	PVBusMapper
CMN700.cmn700_tag_cache.remapper86	PVBusMapper
CMN700.cmn700_tag_cache.remapper87	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.remapper88	PVBusMapper
CMN700.cmn700_tag_cache.remapper89	PVBusMapper
CMN700.cmn700_tag_cache.remapper9	PVBusMapper
CMN700.cmn700_tag_cache.remapper90	PVBusMapper
CMN700.cmn700_tag_cache.remapper91	PVBusMapper
CMN700.cmn700_tag_cache.remapper92	PVBusMapper
CMN700.cmn700_tag_cache.remapper93	PVBusMapper
CMN700.cmn700_tag_cache.remapper94	PVBusMapper
CMN700.cmn700_tag_cache.remapper95	PVBusMapper
CMN700.cmn700_tag_cache.remapper96	PVBusMapper
CMN700.cmn700_tag_cache.remapper97	PVBusMapper
CMN700.cmn700_tag_cache.remapper98	PVBusMapper
CMN700.cmn700_tag_cache.remapper99	PVBusMapper
CMN700.hnf_exclusive_monitor	PVBusExclusiveMonitor
CMN700.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CMN700.hni_exclusive_monitor0	PVBusExclusiveMonitor
CMN700.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CMN700.hni_exclusive_monitor1	PVBusExclusiveMonitor
CMN700.hni_exclusive_monitor1.bus_mapper	PVBusMapper
CMN700.hni_exclusive_monitor2	PVBusExclusiveMonitor
CMN700.hni_exclusive_monitor2.bus_mapper	PVBusMapper
CMN700.hni_exclusive_monitor3	PVBusExclusiveMonitor
CMN700.hni_exclusive_monitor3.bus_mapper	PVBusMapper
CMN700.ocm_decoder	PVBusMapper
CMN700.ocm_exclusive_monitor	PVBusExclusiveMonitor
CMN700.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CMN700.snf_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-894: CMN700 MTI instances

InstanceName	ComponentName
CMN700	CMN700
CMN700.bus_slave_ocm_NS	PVBusSlave
CMN700.bus_slave_ocm_S	PVBusSlave
CMN700.cmn600_cache	CMN600Cache
CMN700.cmn600_cache.upstream[0]	PVBusSlave
CMN700.cmn600_cache.upstream[10]	PVBusSlave
CMN700.cmn600_cache.upstream[11]	PVBusSlave
CMN700.cmn600_cache.upstream[12]	PVBusSlave

InstanceName	ComponentName
CMN700.cmn600_cache.upstream[13]	PVBusSlave
CMN700.cmn600_cache.upstream[14]	PVBusSlave
CMN700.cmn600_cache.upstream[15]	PVBusSlave
CMN700.cmn600_cache.upstream[16]	PVBusSlave
CMN700.cmn600_cache.upstream[17]	PVBusSlave
CMN700.cmn600_cache.upstream[18]	PVBusSlave
CMN700.cmn600_cache.upstream[19]	PVBusSlave
CMN700.cmn600_cache.upstream[1]	PVBusSlave
CMN700.cmn600_cache.upstream[20]	PVBusSlave
CMN700.cmn600_cache.upstream[21]	PVBusSlave
CMN700.cmn600_cache.upstream[22]	PVBusSlave
CMN700.cmn600_cache.upstream[23]	PVBusSlave
CMN700.cmn600_cache.upstream[24]	PVBusSlave
CMN700.cmn600_cache.upstream[25]	PVBusSlave
CMN700.cmn600_cache.upstream[26]	PVBusSlave
CMN700.cmn600_cache.upstream[27]	PVBusSlave
CMN700.cmn600_cache.upstream[28]	PVBusSlave
CMN700.cmn600_cache.upstream[29]	PVBusSlave
CMN700.cmn600_cache.upstream[2]	PVBusSlave
CMN700.cmn600_cache.upstream[30]	PVBusSlave
CMN700.cmn600_cache.upstream[31]	PVBusSlave
CMN700.cmn600_cache.upstream[32]	PVBusSlave
CMN700.cmn600_cache.upstream[33]	PVBusSlave
CMN700.cmn600_cache.upstream[34]	PVBusSlave
CMN700.cmn600_cache.upstream[35]	PVBusSlave
CMN700.cmn600_cache.upstream[36]	PVBusSlave
CMN700.cmn600_cache.upstream[37]	PVBusSlave
CMN700.cmn600_cache.upstream[3]	PVBusSlave
CMN700.cmn600_cache.upstream[4]	PVBusSlave
CMN700.cmn600_cache.upstream[5]	PVBusSlave
CMN700.cmn600_cache.upstream[6]	PVBusSlave
CMN700.cmn600_cache.upstream[7]	PVBusSlave
CMN700.cmn600_cache.upstream[8]	PVBusSlave
CMN700.cmn600_cache.upstream[9]	PVBusSlave
CMN700.cmn700_tag_cache	CMNTAGCACHECADI
CMN700.cmn700_tag_cache.metadata_controller0	ExportTest.CMN700.cmn700_tag_cachemetadata_controller0
CMN700.cmn700_tag_cache.metadata_controller0.MetadataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller1	ExportTest.CMN700.cmn700_tag_cachemetadata_controller1

InstanceName	ComponentName
CMN700.cmn700_tag_cache. metadata_controller1.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller10	ExportTest.CMN700.cmn700_tag_cachemetadata_controller10
CMN700.cmn700_tag_cache. metadata_controller10.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller100	ExportTest.CMN700.cmn700_tag_cachemetadata_controller100
CMN700.cmn700_tag_cache. metadata_controller100.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller101	ExportTest.CMN700.cmn700_tag_cachemetadata_controller101
CMN700.cmn700_tag_cache. metadata_controller101.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller102	ExportTest.CMN700.cmn700_tag_cachemetadata_controller102
CMN700.cmn700_tag_cache. metadata_controller102.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller103	ExportTest.CMN700.cmn700_tag_cachemetadata_controller103
CMN700.cmn700_tag_cache. metadata_controller103.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller104	ExportTest.CMN700.cmn700_tag_cachemetadata_controller104
CMN700.cmn700_tag_cache. metadata_controller104.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller105	ExportTest.CMN700.cmn700_tag_cachemetadata_controller105
CMN700.cmn700_tag_cache. metadata_controller105.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller106	ExportTest.CMN700.cmn700_tag_cachemetadata_controller106
CMN700.cmn700_tag_cache. metadata_controller106.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller107	ExportTest.CMN700.cmn700_tag_cachemetadata_controller107
CMN700.cmn700_tag_cache. metadata_controller107.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller108	ExportTest.CMN700.cmn700_tag_cachemetadata_controller108
CMN700.cmn700_tag_cache. metadata_controller108.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller109	ExportTest.CMN700.cmn700_tag_cachemetadata_controller109
CMN700.cmn700_tag_cache. metadata_controller109.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller11	ExportTest.CMN700.cmn700_tag_cachemetadata_controller11
CMN700.cmn700_tag_cache. metadata_controller11.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller110	ExportTest.CMN700.cmn700_tag_cachemetadata_controller110
CMN700.cmn700_tag_cache. metadata_controller110.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller111	ExportTest.CMN700.cmn700_tag_cachemetadata_controller111
CMN700.cmn700_tag_cache. metadata_controller111.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.metadata_controller112	ExportTest.CMN700.cmn700_tag_cachemetadata_controller112
CMN700.cmn700_tag_cache. metadata_controller112.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller113	ExportTest.CMN700.cmn700_tag_cachemetadata_controller113
CMN700.cmn700_tag_cache. metadata_controller113.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller114	ExportTest.CMN700.cmn700_tag_cachemetadata_controller114
CMN700.cmn700_tag_cache. metadata_controller114.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller115	ExportTest.CMN700.cmn700_tag_cachemetadata_controller115
CMN700.cmn700_tag_cache. metadata_controller115.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller116	ExportTest.CMN700.cmn700_tag_cachemetadata_controller116
CMN700.cmn700_tag_cache. metadata_controller116.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller117	ExportTest.CMN700.cmn700_tag_cachemetadata_controller117
CMN700.cmn700_tag_cache. metadata_controller117.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller118	ExportTest.CMN700.cmn700_tag_cachemetadata_controller118
CMN700.cmn700_tag_cache. metadata_controller118.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller119	ExportTest.CMN700.cmn700_tag_cachemetadata_controller119
CMN700.cmn700_tag_cache. metadata_controller119.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller12	ExportTest.CMN700.cmn700_tag_cachemetadata_controller12
CMN700.cmn700_tag_cache. metadata_controller12.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller120	ExportTest.CMN700.cmn700_tag_cachemetadata_controller120
CMN700.cmn700_tag_cache. metadata_controller120.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller121	ExportTest.CMN700.cmn700_tag_cachemetadata_controller121
CMN700.cmn700_tag_cache. metadata_controller121.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller122	ExportTest.CMN700.cmn700_tag_cachemetadata_controller122
CMN700.cmn700_tag_cache. metadata_controller122.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller123	ExportTest.CMN700.cmn700_tag_cachemetadata_controller123
CMN700.cmn700_tag_cache. metadata_controller123.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller124	ExportTest.CMN700.cmn700_tag_cachemetadata_controller124
CMN700.cmn700_tag_cache. metadata_controller124.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller125	ExportTest.CMN700.cmn700_tag_cachemetadata_controller125

InstanceName	ComponentName
CMN700.cmn700_tag_cache. metadata_controller125.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller126	ExportTest.CMN700.cmn700_tag_cachemetadata_controller126
CMN700.cmn700_tag_cache. metadata_controller126.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller127	ExportTest.CMN700.cmn700_tag_cachemetadata_controller127
CMN700.cmn700_tag_cache. metadata_controller127.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller13	ExportTest.CMN700.cmn700_tag_cachemetadata_controller13
CMN700.cmn700_tag_cache. metadata_controller13.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller14	ExportTest.CMN700.cmn700_tag_cachemetadata_controller14
CMN700.cmn700_tag_cache. metadata_controller14.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller15	ExportTest.CMN700.cmn700_tag_cachemetadata_controller15
CMN700.cmn700_tag_cache. metadata_controller15.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller16	ExportTest.CMN700.cmn700_tag_cachemetadata_controller16
CMN700.cmn700_tag_cache. metadata_controller16.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller17	ExportTest.CMN700.cmn700_tag_cachemetadata_controller17
CMN700.cmn700_tag_cache. metadata_controller17.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller18	ExportTest.CMN700.cmn700_tag_cachemetadata_controller18
CMN700.cmn700_tag_cache. metadata_controller18.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller19	ExportTest.CMN700.cmn700_tag_cachemetadata_controller19
CMN700.cmn700_tag_cache. metadata_controller19.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller2	ExportTest.CMN700.cmn700_tag_cachemetadata_controller2
CMN700.cmn700_tag_cache. metadata_controller2.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller20	ExportTest.CMN700.cmn700_tag_cachemetadata_controller20
CMN700.cmn700_tag_cache. metadata_controller20.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller21	ExportTest.CMN700.cmn700_tag_cachemetadata_controller21
CMN700.cmn700_tag_cache. metadata_controller21.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller22	ExportTest.CMN700.cmn700_tag_cachemetadata_controller22
CMN700.cmn700_tag_cache. metadata_controller22.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller23	ExportTest.CMN700.cmn700_tag_cachemetadata_controller23
CMN700.cmn700_tag_cache. metadata_controller23.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.metadata_controller24	ExportTest.CMN700.cmn700_tag_cachemetadata_controller24
CMN700.cmn700_tag_cache. metadata_controller24.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller25	ExportTest.CMN700.cmn700_tag_cachemetadata_controller25
CMN700.cmn700_tag_cache. metadata_controller25.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller26	ExportTest.CMN700.cmn700_tag_cachemetadata_controller26
CMN700.cmn700_tag_cache. metadata_controller26.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller27	ExportTest.CMN700.cmn700_tag_cachemetadata_controller27
CMN700.cmn700_tag_cache. metadata_controller27.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller28	ExportTest.CMN700.cmn700_tag_cachemetadata_controller28
CMN700.cmn700_tag_cache. metadata_controller28.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller29	ExportTest.CMN700.cmn700_tag_cachemetadata_controller29
CMN700.cmn700_tag_cache. metadata_controller29.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller3	ExportTest.CMN700.cmn700_tag_cachemetadata_controller3
CMN700.cmn700_tag_cache. metadata_controller3.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller30	ExportTest.CMN700.cmn700_tag_cachemetadata_controller30
CMN700.cmn700_tag_cache. metadata_controller30.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller31	ExportTest.CMN700.cmn700_tag_cachemetadata_controller31
CMN700.cmn700_tag_cache. metadata_controller31.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller32	ExportTest.CMN700.cmn700_tag_cachemetadata_controller32
CMN700.cmn700_tag_cache. metadata_controller32.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller33	ExportTest.CMN700.cmn700_tag_cachemetadata_controller33
CMN700.cmn700_tag_cache. metadata_controller33.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller34	ExportTest.CMN700.cmn700_tag_cachemetadata_controller34
CMN700.cmn700_tag_cache. metadata_controller34.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller35	ExportTest.CMN700.cmn700_tag_cachemetadata_controller35
CMN700.cmn700_tag_cache. metadata_controller35.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller36	ExportTest.CMN700.cmn700_tag_cachemetadata_controller36
CMN700.cmn700_tag_cache. metadata_controller36.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller37	ExportTest.CMN700.cmn700_tag_cachemetadata_controller37

InstanceName	ComponentName
CMN700.cmn700_tag_cache. metadata_controller37.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller38	ExportTest.CMN700.cmn700_tag_cachemetadata_controller38
CMN700.cmn700_tag_cache. metadata_controller38.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller39	ExportTest.CMN700.cmn700_tag_cachemetadata_controller39
CMN700.cmn700_tag_cache. metadata_controller39.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller4	ExportTest.CMN700.cmn700_tag_cachemetadata_controller4
CMN700.cmn700_tag_cache. metadata_controller4.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller40	ExportTest.CMN700.cmn700_tag_cachemetadata_controller40
CMN700.cmn700_tag_cache. metadata_controller40.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller41	ExportTest.CMN700.cmn700_tag_cachemetadata_controller41
CMN700.cmn700_tag_cache. metadata_controller41.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller42	ExportTest.CMN700.cmn700_tag_cachemetadata_controller42
CMN700.cmn700_tag_cache. metadata_controller42.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller43	ExportTest.CMN700.cmn700_tag_cachemetadata_controller43
CMN700.cmn700_tag_cache. metadata_controller43.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller44	ExportTest.CMN700.cmn700_tag_cachemetadata_controller44
CMN700.cmn700_tag_cache. metadata_controller44.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller45	ExportTest.CMN700.cmn700_tag_cachemetadata_controller45
CMN700.cmn700_tag_cache. metadata_controller45.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller46	ExportTest.CMN700.cmn700_tag_cachemetadata_controller46
CMN700.cmn700_tag_cache. metadata_controller46.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller47	ExportTest.CMN700.cmn700_tag_cachemetadata_controller47
CMN700.cmn700_tag_cache. metadata_controller47.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller48	ExportTest.CMN700.cmn700_tag_cachemetadata_controller48
CMN700.cmn700_tag_cache. metadata_controller48.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller49	ExportTest.CMN700.cmn700_tag_cachemetadata_controller49
CMN700.cmn700_tag_cache. metadata_controller49.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller5	ExportTest.CMN700.cmn700_tag_cachemetadata_controller5
CMN700.cmn700_tag_cache. metadata_controller5.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.metadata_controller50	ExportTest.CMN700.cmn700_tag_cachemetadata_controller50
CMN700.cmn700_tag_cache. metadata_controller50.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller51	ExportTest.CMN700.cmn700_tag_cachemetadata_controller51
CMN700.cmn700_tag_cache. metadata_controller51.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller52	ExportTest.CMN700.cmn700_tag_cachemetadata_controller52
CMN700.cmn700_tag_cache. metadata_controller52.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller53	ExportTest.CMN700.cmn700_tag_cachemetadata_controller53
CMN700.cmn700_tag_cache. metadata_controller53.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller54	ExportTest.CMN700.cmn700_tag_cachemetadata_controller54
CMN700.cmn700_tag_cache. metadata_controller54.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller55	ExportTest.CMN700.cmn700_tag_cachemetadata_controller55
CMN700.cmn700_tag_cache. metadata_controller55.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller56	ExportTest.CMN700.cmn700_tag_cachemetadata_controller56
CMN700.cmn700_tag_cache. metadata_controller56.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller57	ExportTest.CMN700.cmn700_tag_cachemetadata_controller57
CMN700.cmn700_tag_cache. metadata_controller57.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller58	ExportTest.CMN700.cmn700_tag_cachemetadata_controller58
CMN700.cmn700_tag_cache. metadata_controller58.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller59	ExportTest.CMN700.cmn700_tag_cachemetadata_controller59
CMN700.cmn700_tag_cache. metadata_controller59.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller6	ExportTest.CMN700.cmn700_tag_cachemetadata_controller6
CMN700.cmn700_tag_cache. metadata_controller6.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller60	ExportTest.CMN700.cmn700_tag_cachemetadata_controller60
CMN700.cmn700_tag_cache. metadata_controller60.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller61	ExportTest.CMN700.cmn700_tag_cachemetadata_controller61
CMN700.cmn700_tag_cache. metadata_controller61.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller62	ExportTest.CMN700.cmn700_tag_cachemetadata_controller62
CMN700.cmn700_tag_cache. metadata_controller62.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller63	ExportTest.CMN700.cmn700_tag_cachemetadata_controller63

InstanceName	ComponentName
CMN700.cmn700_tag_cache. metadata_controller63.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller64	ExportTest.CMN700.cmn700_tag_cachemetadata_controller64
CMN700.cmn700_tag_cache. metadata_controller64.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller65	ExportTest.CMN700.cmn700_tag_cachemetadata_controller65
CMN700.cmn700_tag_cache. metadata_controller65.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller66	ExportTest.CMN700.cmn700_tag_cachemetadata_controller66
CMN700.cmn700_tag_cache. metadata_controller66.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller67	ExportTest.CMN700.cmn700_tag_cachemetadata_controller67
CMN700.cmn700_tag_cache. metadata_controller67.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller68	ExportTest.CMN700.cmn700_tag_cachemetadata_controller68
CMN700.cmn700_tag_cache. metadata_controller68.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller69	ExportTest.CMN700.cmn700_tag_cachemetadata_controller69
CMN700.cmn700_tag_cache. metadata_controller69.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller7	ExportTest.CMN700.cmn700_tag_cachemetadata_controller7
CMN700.cmn700_tag_cache. metadata_controller7.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller70	ExportTest.CMN700.cmn700_tag_cachemetadata_controller70
CMN700.cmn700_tag_cache. metadata_controller70.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller71	ExportTest.CMN700.cmn700_tag_cachemetadata_controller71
CMN700.cmn700_tag_cache. metadata_controller71.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller72	ExportTest.CMN700.cmn700_tag_cachemetadata_controller72
CMN700.cmn700_tag_cache. metadata_controller72.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller73	ExportTest.CMN700.cmn700_tag_cachemetadata_controller73
CMN700.cmn700_tag_cache. metadata_controller73.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller74	ExportTest.CMN700.cmn700_tag_cachemetadata_controller74
CMN700.cmn700_tag_cache. metadata_controller74.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller75	ExportTest.CMN700.cmn700_tag_cachemetadata_controller75
CMN700.cmn700_tag_cache. metadata_controller75.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller76	ExportTest.CMN700.cmn700_tag_cachemetadata_controller76
CMN700.cmn700_tag_cache. metadata_controller76.MetaDataMapper	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.metadata_controller77	ExportTest.CMN700.cmn700_tag_cachemetadata_controller77
CMN700.cmn700_tag_cache. metadata_controller77.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller78	ExportTest.CMN700.cmn700_tag_cachemetadata_controller78
CMN700.cmn700_tag_cache. metadata_controller78.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller79	ExportTest.CMN700.cmn700_tag_cachemetadata_controller79
CMN700.cmn700_tag_cache. metadata_controller79.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller8	ExportTest.CMN700.cmn700_tag_cachemetadata_controller8
CMN700.cmn700_tag_cache. metadata_controller8.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller80	ExportTest.CMN700.cmn700_tag_cachemetadata_controller80
CMN700.cmn700_tag_cache. metadata_controller80.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller81	ExportTest.CMN700.cmn700_tag_cachemetadata_controller81
CMN700.cmn700_tag_cache. metadata_controller81.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller82	ExportTest.CMN700.cmn700_tag_cachemetadata_controller82
CMN700.cmn700_tag_cache. metadata_controller82.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller83	ExportTest.CMN700.cmn700_tag_cachemetadata_controller83
CMN700.cmn700_tag_cache. metadata_controller83.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller84	ExportTest.CMN700.cmn700_tag_cachemetadata_controller84
CMN700.cmn700_tag_cache. metadata_controller84.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller85	ExportTest.CMN700.cmn700_tag_cachemetadata_controller85
CMN700.cmn700_tag_cache. metadata_controller85.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller86	ExportTest.CMN700.cmn700_tag_cachemetadata_controller86
CMN700.cmn700_tag_cache. metadata_controller86.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller87	ExportTest.CMN700.cmn700_tag_cachemetadata_controller87
CMN700.cmn700_tag_cache. metadata_controller87.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller88	ExportTest.CMN700.cmn700_tag_cachemetadata_controller88
CMN700.cmn700_tag_cache. metadata_controller88.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller89	ExportTest.CMN700.cmn700_tag_cachemetadata_controller89
CMN700.cmn700_tag_cache. metadata_controller89.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller9	ExportTest.CMN700.cmn700_tag_cachemetadata_controller9

InstanceName	ComponentName
CMN700.cmn700_tag_cache. metadata_controller9.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller90	ExportTest.CMN700.cmn700_tag_cachemetadata_controller90
CMN700.cmn700_tag_cache. metadata_controller90.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller91	ExportTest.CMN700.cmn700_tag_cachemetadata_controller91
CMN700.cmn700_tag_cache. metadata_controller91.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller92	ExportTest.CMN700.cmn700_tag_cachemetadata_controller92
CMN700.cmn700_tag_cache. metadata_controller92.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller93	ExportTest.CMN700.cmn700_tag_cachemetadata_controller93
CMN700.cmn700_tag_cache. metadata_controller93.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller94	ExportTest.CMN700.cmn700_tag_cachemetadata_controller94
CMN700.cmn700_tag_cache. metadata_controller94.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller95	ExportTest.CMN700.cmn700_tag_cachemetadata_controller95
CMN700.cmn700_tag_cache. metadata_controller95.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller96	ExportTest.CMN700.cmn700_tag_cachemetadata_controller96
CMN700.cmn700_tag_cache. metadata_controller96.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller97	ExportTest.CMN700.cmn700_tag_cachemetadata_controller97
CMN700.cmn700_tag_cache. metadata_controller97.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller98	ExportTest.CMN700.cmn700_tag_cachemetadata_controller98
CMN700.cmn700_tag_cache. metadata_controller98.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.metadata_controller99	ExportTest.CMN700.cmn700_tag_cachemetadata_controller99
CMN700.cmn700_tag_cache. metadata_controller99.MetaDataMapper	PVBusMapper
CMN700.cmn700_tag_cache.remapper0	PVBusMapper
CMN700.cmn700_tag_cache.remapper1	PVBusMapper
CMN700.cmn700_tag_cache.remapper10	PVBusMapper
CMN700.cmn700_tag_cache.remapper100	PVBusMapper
CMN700.cmn700_tag_cache.remapper101	PVBusMapper
CMN700.cmn700_tag_cache.remapper102	PVBusMapper
CMN700.cmn700_tag_cache.remapper103	PVBusMapper
CMN700.cmn700_tag_cache.remapper104	PVBusMapper
CMN700.cmn700_tag_cache.remapper105	PVBusMapper
CMN700.cmn700_tag_cache.remapper106	PVBusMapper
CMN700.cmn700_tag_cache.remapper107	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.remapper108	PVBusMapper
CMN700.cmn700_tag_cache.remapper109	PVBusMapper
CMN700.cmn700_tag_cache.remapper11	PVBusMapper
CMN700.cmn700_tag_cache.remapper110	PVBusMapper
CMN700.cmn700_tag_cache.remapper111	PVBusMapper
CMN700.cmn700_tag_cache.remapper112	PVBusMapper
CMN700.cmn700_tag_cache.remapper113	PVBusMapper
CMN700.cmn700_tag_cache.remapper114	PVBusMapper
CMN700.cmn700_tag_cache.remapper115	PVBusMapper
CMN700.cmn700_tag_cache.remapper116	PVBusMapper
CMN700.cmn700_tag_cache.remapper117	PVBusMapper
CMN700.cmn700_tag_cache.remapper118	PVBusMapper
CMN700.cmn700_tag_cache.remapper119	PVBusMapper
CMN700.cmn700_tag_cache.remapper12	PVBusMapper
CMN700.cmn700_tag_cache.remapper120	PVBusMapper
CMN700.cmn700_tag_cache.remapper121	PVBusMapper
CMN700.cmn700_tag_cache.remapper122	PVBusMapper
CMN700.cmn700_tag_cache.remapper123	PVBusMapper
CMN700.cmn700_tag_cache.remapper124	PVBusMapper
CMN700.cmn700_tag_cache.remapper125	PVBusMapper
CMN700.cmn700_tag_cache.remapper126	PVBusMapper
CMN700.cmn700_tag_cache.remapper127	PVBusMapper
CMN700.cmn700_tag_cache.remapper13	PVBusMapper
CMN700.cmn700_tag_cache.remapper14	PVBusMapper
CMN700.cmn700_tag_cache.remapper15	PVBusMapper
CMN700.cmn700_tag_cache.remapper16	PVBusMapper
CMN700.cmn700_tag_cache.remapper17	PVBusMapper
CMN700.cmn700_tag_cache.remapper18	PVBusMapper
CMN700.cmn700_tag_cache.remapper19	PVBusMapper
CMN700.cmn700_tag_cache.remapper2	PVBusMapper
CMN700.cmn700_tag_cache.remapper20	PVBusMapper
CMN700.cmn700_tag_cache.remapper21	PVBusMapper
CMN700.cmn700_tag_cache.remapper22	PVBusMapper
CMN700.cmn700_tag_cache.remapper23	PVBusMapper
CMN700.cmn700_tag_cache.remapper24	PVBusMapper
CMN700.cmn700_tag_cache.remapper25	PVBusMapper
CMN700.cmn700_tag_cache.remapper26	PVBusMapper
CMN700.cmn700_tag_cache.remapper27	PVBusMapper
CMN700.cmn700_tag_cache.remapper28	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.remapper29	PVBusMapper
CMN700.cmn700_tag_cache.remapper3	PVBusMapper
CMN700.cmn700_tag_cache.remapper30	PVBusMapper
CMN700.cmn700_tag_cache.remapper31	PVBusMapper
CMN700.cmn700_tag_cache.remapper32	PVBusMapper
CMN700.cmn700_tag_cache.remapper33	PVBusMapper
CMN700.cmn700_tag_cache.remapper34	PVBusMapper
CMN700.cmn700_tag_cache.remapper35	PVBusMapper
CMN700.cmn700_tag_cache.remapper36	PVBusMapper
CMN700.cmn700_tag_cache.remapper37	PVBusMapper
CMN700.cmn700_tag_cache.remapper38	PVBusMapper
CMN700.cmn700_tag_cache.remapper39	PVBusMapper
CMN700.cmn700_tag_cache.remapper4	PVBusMapper
CMN700.cmn700_tag_cache.remapper40	PVBusMapper
CMN700.cmn700_tag_cache.remapper41	PVBusMapper
CMN700.cmn700_tag_cache.remapper42	PVBusMapper
CMN700.cmn700_tag_cache.remapper43	PVBusMapper
CMN700.cmn700_tag_cache.remapper44	PVBusMapper
CMN700.cmn700_tag_cache.remapper45	PVBusMapper
CMN700.cmn700_tag_cache.remapper46	PVBusMapper
CMN700.cmn700_tag_cache.remapper47	PVBusMapper
CMN700.cmn700_tag_cache.remapper48	PVBusMapper
CMN700.cmn700_tag_cache.remapper49	PVBusMapper
CMN700.cmn700_tag_cache.remapper5	PVBusMapper
CMN700.cmn700_tag_cache.remapper50	PVBusMapper
CMN700.cmn700_tag_cache.remapper51	PVBusMapper
CMN700.cmn700_tag_cache.remapper52	PVBusMapper
CMN700.cmn700_tag_cache.remapper53	PVBusMapper
CMN700.cmn700_tag_cache.remapper54	PVBusMapper
CMN700.cmn700_tag_cache.remapper55	PVBusMapper
CMN700.cmn700_tag_cache.remapper56	PVBusMapper
CMN700.cmn700_tag_cache.remapper57	PVBusMapper
CMN700.cmn700_tag_cache.remapper58	PVBusMapper
CMN700.cmn700_tag_cache.remapper59	PVBusMapper
CMN700.cmn700_tag_cache.remapper6	PVBusMapper
CMN700.cmn700_tag_cache.remapper60	PVBusMapper
CMN700.cmn700_tag_cache.remapper61	PVBusMapper
CMN700.cmn700_tag_cache.remapper62	PVBusMapper
CMN700.cmn700_tag_cache.remapper63	PVBusMapper

InstanceName	ComponentName
CMN700.cmn700_tag_cache.remapper64	PVBusMapper
CMN700.cmn700_tag_cache.remapper65	PVBusMapper
CMN700.cmn700_tag_cache.remapper66	PVBusMapper
CMN700.cmn700_tag_cache.remapper67	PVBusMapper
CMN700.cmn700_tag_cache.remapper68	PVBusMapper
CMN700.cmn700_tag_cache.remapper69	PVBusMapper
CMN700.cmn700_tag_cache.remapper7	PVBusMapper
CMN700.cmn700_tag_cache.remapper70	PVBusMapper
CMN700.cmn700_tag_cache.remapper71	PVBusMapper
CMN700.cmn700_tag_cache.remapper72	PVBusMapper
CMN700.cmn700_tag_cache.remapper73	PVBusMapper
CMN700.cmn700_tag_cache.remapper74	PVBusMapper
CMN700.cmn700_tag_cache.remapper75	PVBusMapper
CMN700.cmn700_tag_cache.remapper76	PVBusMapper
CMN700.cmn700_tag_cache.remapper77	PVBusMapper
CMN700.cmn700_tag_cache.remapper78	PVBusMapper
CMN700.cmn700_tag_cache.remapper79	PVBusMapper
CMN700.cmn700_tag_cache.remapper8	PVBusMapper
CMN700.cmn700_tag_cache.remapper80	PVBusMapper
CMN700.cmn700_tag_cache.remapper81	PVBusMapper
CMN700.cmn700_tag_cache.remapper82	PVBusMapper
CMN700.cmn700_tag_cache.remapper83	PVBusMapper
CMN700.cmn700_tag_cache.remapper84	PVBusMapper
CMN700.cmn700_tag_cache.remapper85	PVBusMapper
CMN700.cmn700_tag_cache.remapper86	PVBusMapper
CMN700.cmn700_tag_cache.remapper87	PVBusMapper
CMN700.cmn700_tag_cache.remapper88	PVBusMapper
CMN700.cmn700_tag_cache.remapper89	PVBusMapper
CMN700.cmn700_tag_cache.remapper9	PVBusMapper
CMN700.cmn700_tag_cache.remapper90	PVBusMapper
CMN700.cmn700_tag_cache.remapper91	PVBusMapper
CMN700.cmn700_tag_cache.remapper92	PVBusMapper
CMN700.cmn700_tag_cache.remapper93	PVBusMapper
CMN700.cmn700_tag_cache.remapper94	PVBusMapper
CMN700.cmn700_tag_cache.remapper95	PVBusMapper
CMN700.cmn700_tag_cache.remapper96	PVBusMapper
CMN700.cmn700_tag_cache.remapper97	PVBusMapper
CMN700.cmn700_tag_cache.remapper98	PVBusMapper
CMN700.cmn700_tag_cache.remapper99	PVBusMapper

InstanceName	ComponentName
CMN700.hnf_exclusive_monitor	PVBusExclusiveMonitor
CMN700.hnf_exclusive_monitor.bus_mapper	PVBusMapper
CMN700.hni_exclusive_monitor0	PVBusExclusiveMonitor
CMN700.hni_exclusive_monitor0.bus_mapper	PVBusMapper
CMN700.hni_exclusive_monitor1	PVBusExclusiveMonitor
CMN700.hni_exclusive_monitor1.bus_mapper	PVBusMapper
CMN700.hni_exclusive_monitor2	PVBusExclusiveMonitor
CMN700.hni_exclusive_monitor2.bus_mapper	PVBusMapper
CMN700.hni_exclusive_monitor3	PVBusExclusiveMonitor
CMN700.hni_exclusive_monitor3.bus_mapper	PVBusMapper
CMN700.ocm_decoder	PVBusMapper
CMN700.ocm_exclusive_monitor	PVBusExclusiveMonitor
CMN700.ocm_exclusive_monitor.bus_mapper	PVBusMapper
CMN700.snf_mapper	PVBusMapper

CMN700 contains the following CADI targets:

- CMN700
- CMN_TAG_CACHE

About the model

- For information about the changes in this release, see the Fast Models Portfolio release notes.
- Major IP revisions (rX) are modeled and are controlled by the `revision` parameter or the topology file.
- Minor IP revisions (pY) are not modeled. There are no functional differences in the model between pY revisions, with the exception of the `periph_id2` register.
- To configure the model, you must have installed Arm® Socrates™. The `mesh_config_file` parameter defines the mesh placement of the CHI nodes. Set it to the name of the yaml configuration file emitted by Socrates. You must use version SYSOC-BN-00001 r1p6-02lac1 of Socrates to generate the file. For more information about Socrates, see [Arm Socrates](#) on Arm Developer. To download it, contact [Arm Technical Support](#).
- Interconnect models are based on the TRM description and do not typically model RTL defects.
- The maximum mesh size supported is X=12, Y=12.
- The mapping between the port number for rnf, rni/rnd, hni, and snf/sbsx interface ports and `nodeId` is based on the `nodeId` index. For example, RNF2 controls `pvbus_s_rnf[2]`. Its index is specified in the `node_info` register as `logical_id`. Similar behavior can be expected for HN-I.

If both RN-D and RN-I nodes are present, then all starting rni ports are mapped to RN-D nodes and then the RN-I nodes. For example, for CMN700 with two RN-D nodes, one RN-I node, and given each RN-I or RN-D node controls three interface ports, `pvbus_s_rni[0-2]` maps to RND0, `pvbus_s_rni[3-5]` maps to RND1 and `pvbus_s_rni[6-8]` maps to RNI0.

Similarly, SN-F and SBSX nodes are mapped to `pvbuss_m_snf[]` ports where starting ports are mapped to SN-F and then SBSX nodes.

- See the CMN700r3 TRM for the `HN*_SLC_SIZE_PARAM` values.
- There is limited support for RNSAMs external to the CMN. See the Model limitations section for more information.
- CCG device id is CCG id + 1 when CAL2 and PCIE_ENABLE are set for port 1.
- CXL Type-3 (CXL.mem) devices can be connected to the `pvbuss_m_cxs[]` ports. The CXRH nodes, if any, are connected to `pvbuss_m_cxs[0]` ... followed by the CCG nodes, if any.
- The Address Based Flush (ABF) feature is available. After it has started, the ABF completes instantly. The ABF implementation has the following limitations:
 - In the Fast Model, any HN-F in the interconnect can perform an ABF on any address range even if that range is not part of the System Cache Group (SCG) that the HN-F belongs to.
 - Clearing the OCM with an ABF is not supported. A trace has been added to highlight this.
 - When `HN*_ABF_PR.abf_mode` is Reserved and `HN*_ABF_PR.abf_enable` is set, an error is printed and the ABF status register is set to complete successfully, but no operation takes place.
 - Because an ABF completes instantly, the `SF_PM_TRANSITION_ABORT` state cannot be reached. Contact [Arm Technical Support](#) with questions about support for this error condition.
 - The optional interrupt `INTREQPPU` is not supported.
- There is limited support for RAS:
 - Error logging and reporting functionality for HN-I, SBSX, XP, and MTU are supported.
 - RAS-related interrupts (`INTREQERRS`, `INTREQFAULTS`, `INTREQFAULTNS`, `INTREQERRNS`) have been added.
 - Central RAS interrupt-handling functionality of HN-D is supported.



Enabling RAS can impact performance. To avoid this, RAS is disabled by default in the CMN700 model. To enable RAS, set the `enable_ras` parameter to true.

-
- There is support for A4S Multichip routing, with limitations:
 - When the `enable_a4s` parameter is false, top-level model ports are terminated with abort handlers.
 - Set `enable_a4s` parameter to true to opt into the feature.
 - The model routes A4S transactions from the `ic_dr_a4s` port to `tx_cxs_a4s` ports according to A4S LDIDs.
 - There is 1 A4S tx/rx port for each CCG up to a maximum of 32 CCGs.
 - When the remote transaction arrives at the receiving remote CMN on `rx_cxs_a4s` ports, it is routed to the GIC A4S port (`ic_rd_a4s`).

- The A4S LDID for the CCGs can be found by reading the `ccg_RA.unit_info` register or through model parameter `print_cmn_config`.

A4S support limitations are listed in the Model Limitations section.

The following table describes the level of support in the CMN700 model for different revision-specific features of the IP:

Table 3-895: Model support for revision-specific features

Revision	Feature	Model support
r1p0, second release	CXLv2.0 host-side support for CXL.mem and CXL.io protocols for Type-3 memory expansion devices	Model supports Type-3 connections using PVBUS
	32 CCG or CXG gateway nodes	Supported
	Non power-of-2 hashing of HN-Fs with $2N * \{1, 3, \text{or } 5\}$ up to 64 HN-Fs or 128 HN-Fs with CAL	Supported
r2p0, third release	Remote PCIe streaming support	Not in scope
	1.5MB SLC support	Supported: SLC_SIZE=2, NUM_WAYS=12
	90 RN-I support	Only 40 supported (3 AXI port each)
	128 SN-F/SBSX support	Only 80 supported
	AXID based for port aggregation across chip	Not supported
	RNSAM support for 4 chip flat hashing configuration	Supported
r3p0, fourth release	AXU port on all MXPs	Not supported
	512 RN-I requests support	Not in scope
	16-bit REQ RSVDC support	Width reported in info_global register; RSVDC not in scope
	Configurable write cancel threshold in RN-I and RN-D	Not in scope
	Remote DVM sync collapsing	Not in scope
	CPAG MOD-3 hashing	CPAG hashing not supported
	PCIe write streaming improvements	Not in scope
r3p3, fifth release	Performance optimization guideline improvements for RN-I and RN-D	Not in scope
	HN-P and HN-I AxID Encoding improvements	Not in scope

Model limitations

- Out of scope:
 - PMU counters are not supported. Counter registers are implemented as RAZ.
 - QoS is not supported and all related registers are RAZ/WI.
 - Error injection and error generation are not supported. All error registers are RAZ/WI.
 - Power, clock, and interrupt signals are not supported, but RAS-related interrupts are supported.

The P-channel (power) signals `PREQ_LOGIC`, `PSTATE_LOGIC`, `PACCEPT_LOGIC`, `PDENY_LOGIC`, and `PACTIVE_LOGIC`, are not implemented. The model behaves as if `PSTATE_LOGIC` is 5'b00000. So the initial power state of the HN-F nodes is NOSFSLC/OFF. This is reflected in the reset values of `cmn_hns_ppu_pwpr`, which was named

`por_hnf_ppu_pwpr` in earlier IP revisions. Software must write the `*_ppu_pwpr` registers to change the power state. Both HAM and FAM power states enable all of the SLC ways.

- Protocol credits and flit buffers are not supported.
- Snooper filtering is not supported.
- Prefetch target operations are not supported.
- The MTE feature has been minimally tested and the functionality cannot be guaranteed. MTE error injection and detection are not supported.
- An access to a reserved or nonexistent register does not abort. It returns `pv::Tx_Data::TX_OK` and is RAZ/WI.
- Topologies with three I/O masters connected through an RN-I bridge device have been tested. Configurations where multiple I/O masters share a single ACE-lite port have not been tested.
- RNSAMs external to the mesh network are not functionally supported.



Note

Model parameter `force_rnsam_internal` is true by default. If this parameter is set to false and the topology has an RNFx (non-ESAM), the external bit (bit[31]) of the RNSAM children pointer is enabled. This bit is the only functionality governed by this parameter.

- By default, HN-F hashing uses the address [MAX:12] instead of the actual address [MAX:6], due to the DMI mechanism in the model. Enable the parameter `enable_rnsam_to_hnf_wider_hash` to make hashing logic use the actual address [MAX:6], but this might reduce the simulation speed of the model.
- OKB SLC data RAM and tag RAM are not supported.
- Source-based SLC cache partitioning is not supported.
- Way-based SLC cache partitioning is not supported.
- SLC/LCC partitioning dynamic allocation is not supported.
- Remote chip addresses incorrectly allocate a line in the SLC when `cache_state_modelled=true`. This can cause unexpected cache behavior that does not match the RTL, exhausting the SLC earlier or extra evictions.
- The fields in the `mxp_device_port_connect_info` register corresponding to the number of device credit slices and the number of CAL credit slices are not supported.
- HN-Fs with different SLC sizes in the same configuration are not supported.
- GIC communication over A4S ports is not supported.
- No support or updates for the following parameters:
 - `POR_RSVD_STRONGNC_EN_PARAM`
 - `POR_HNSAM_CUSTOM_REGS_PARAM`
- No updates for a new bit in `CMN_HNS_CFG_CTL` to disable HNS stashing snoop (`hnf_stash_snp_dis`).
- HND-APB registers not supported.
- HN-P nodes are not supported as hashed target from the RNSAM.

- There is limited support for CXL Type-3. It only supports a single device connection (`sa_ports_cnt`).
- For CMN700R1, `por_hnf_cfg_ctl` follows the CMN700R0 write mask and reset value.
- For CMN700R1 and later, stash snooping is not supported.
- The model cannot activate both CCG APB register access traces and CMN register access traces simultaneously. Use the parameter `register_traces_for_ccg_apb_accesses` to enable CCG APB register access traces. By default, CMN register access traces are available for activation.
- MPAM features are not supported in the model, and the configuration-dependent register reset values do not match the RTL. Some of the registers (`por_hnf_mpam_idr` to `por_hnf_mpam_mbwumon_idr`) shared between security modes are not present in MPAM_S.
- The MPAM reset values described in `por_hnf_mpam_ns_por_hnf_mpam_idr[31:24]` default to the not-supported state. The model parameter `hnf_mpam_idr_override` can be used to override this but does not implement any of the functionality.
- The following limitations apply to System Cache Groups and Hash Target Groups:
 - The `hashed_target_grp_hnp_nodeid_reg` which is used for CCG/HN-P target IDs is not supported.
 - AXID hashing across HN-P/CCGs is not supported.
 - HTGs containing targets across multi-chips are not supported.
- Multiple CPA groups are not supported.
- MPAM_S `secure_register_groups_override` is not supported.
- No specific Debug Trace (DT) node functionality is supported. See [4. Plug-ins for Fast Models](#) on page 4138 for information about using plug-ins for trace.
- For Power State Transition related flush/disable of the SLC due to HNF_PPU_PWPR register, only the Operational Mode field (bits[7:4]) is considered. The Power Policy field (bits[3:0]) is ignored.
- Setting HAM mode for the Operational Mode field (bits[7:4]) of the HNF_PPU_PWPR is equivalent to FAM mode. The SLCH2 is neither flushed nor disabled according to the Power Policy field (bits[3:0]).
- Only hashing granularities of 4096B and above are supported.
- All RNs are assumed to have the same HN mapping. Programming RNSAMs differently between RNs is not supported.
- RNSAM cannot be programmed with the same HN-F target in a hashed and non-hashed memory region.
- `RNSAM_STATUS.default_target_type` and `RNSAM_STATUS.default_nodeid` are used as the default target only when `RNSAM_STATUS.use_default_node=1`. Otherwise, when a `txn` is sent into the model that does not belong to any of the address regions programmed in the RNSAM, it is routed to the HN-D irrespective of what is programmed in the `RNSAM_STATUS.default_target_type` and `RNSAM_STATUS.default_nodeid` fields.
- Address bit masking is not supported in RNSAM or HN-F SAM.

- The OCM does not keep track of the total memory used across the S/NS security world and does not perform any eviction either. It is up to the software to make sure it does not oversubscribe the OCM.
- RSVDC StrongNC and its associated functionality is not supported.
- User-defined hashing mechanism in an SCG is not supported.
- The CXSA mode has limited support. Currently, it only supports one aggregated device.
- The model does not display any register traces.
- GenericTrace for the CMN700 Fast Model, incorrectly mentions "CMN600" in logs.
- CCG node addresses do not match RTL node addresses if not using node addresses from yml.
- The following limitations are specific to revision r2p0:
 - The model does not support the RA_PRESENT configurable option. RAs are always present in CCG.
 - Maximum number of RN-D supported is 40.
- The following limitations apply to revisions r2p0 and r3p0:
 - No support for RWL (ReadWriteLock).
 - Maximum number of RN-I supported is 40.
 - Maximum combined number of RN-I and RN-D is 40.
 - SN-Fs on CAL4 are not supported.
 - Maximum number of SN interfaces supported is 80.
- The following limitations are specific to revision r3p0:
 - Hybrid CAL flavors are not supported.
 - Maximum RAID of 1024 is not supported.
 - Direct Subordinate Access (DSA) CCG inbound request bypass of HN-F is not supported.
 - CXL v2.0 device support for various types is not supported or verified.
 - CXL v2.0 host support for various types is not supported or verified.
 - There is no support for CPAG MOD-3 hashing.
 - There is no support for AXU.
- RAS feature limitations:
 - Error logging and reporting functionality for CCG, HN-F, and CXHA are not supported.
 - Single-bit error injection for MTU is not supported as there is no ECC checker or register present to support it.
 - NDE response and Poison error check are not supported.
 - Flit parity and Data check errors are not supported.
 - The information that is captured as source ID, target ID, and logical ID in the ERRMISC register might not be correct or match the hardware.
 - HN-D Illegal Configuration check does not check that the access is of device type.

- HN-D Illegal Configuration check does not check the access security mode.
- SN-F RAS errors are treated as SBSX errors.
- A4S support limitations:
 - GIC_DESTID input strap is not supported. Incoming transactions from remote chip are always routed to IC_RD.
 - The model assumes the presence of 1x A4S port for GIC without regard for the actual number of a4s interfaces in `thesh_config_file` topology.
 - The model does not require user software at runtime to enable the CMN to route multichip A4S transactions between chips.
 - Limited performance testing has been performed.
 - There is no support for the use of "id_map" file specified by CMN Configuration Integration Manual (CIM) to configure the model for reset.
- Model behavior does not reflect errata notice 2732981. The model behaves as r3p1. See the errata for details.
- For an HTG/SCG, the model does not consider the secondary region if the primary region is not valid.
- HN-D is only permitted on device port P2 in a single-MXP configuration.
- Only the last RN-F in the mesh, which is the one with the highest logical id, can be controlled by `mxp_p[0-5]_syscoreq_ctl` registers. Also, incorrectly, it can be controlled from any XP.
- HNSAM only supports two non-hashed memory regions. Memory regions programmed using `cmn_hns_sam_nonhash_cfg[1|2]_memregion2-63` registers are ignored.
- The Hierarchical hashing fields `HASHED_TARGET_GRP_HASH_CNTL_REG.htg_region#{index}_hier_enable_address_striping` and `HASHED_TARGET_GRP_HASH_CNTL_REG.htg_region#{index}_hier_cluster_mask` are not supported.
- The `HNSAM_DEF_HASHED_GRP_EN` yml parameter is not supported and HN-F SAM legacy mode is always enabled.
- The `HNSAM_NUM_HTG` yml parameter is not supported.
- When both `POR_CCLA_ULL_CTL.ull_to_ull_en` and `POR_CCLA_ULL_CTL.send_vd_init` bits are set then both `POR_CCLA_ULL_STATUS.tx_state` and `POR_CCLA_ULL_STATUS.rx_state` are set. The other side of the link is not consulted to set `POR_CCLA_ULL_STATUS.rx_state`.
- HN-F SAM:
 - Address masking in default hash regions in HN-F SAM is not supported.
 - Hashing in default hash regions in HN-F SAM is not supported. However, trace `SNF_HASHING_Target_SNF` displays the target SNF as if 3SN/6SN hashing were enabled.



The transaction is always routed to the SN0 nodeid programmed in the SAM_CONTROL register.

- Re-programming regions in HN-F SAM is not tested.
- Hashing across CCGs in HN-F SAM is not supported.
- Address compare hashed regions in HN-F SAM do not support non-power of 2 hashing. When non-power of 2 hashing is enabled, the first SN in the HTG (SN0) is used as the target.
- HNSAM_DEF_HASHED_GRP_EN yml parameter is not supported.
- CMN700 r1p0 supports only 8 hashed regions in HN-F SAM. CMN700 r2p0 and r3p0 support 16 hashed regions.
- Default hash regions in HN-F SAM have limited support.
`cmn_hns_sam_cfg1_def_hashed_region` and `cmn_hns_sam_cfg2_def_hashed_region` are not supported.
- Hashing across SN-F on CAL2 in HN-F SAM is not supported.

About the `debug_force_snoop` parameter

The CMN interconnect normally starts up with snooping disabled.

The parameters `rnf_sci_enable` and `rni_sci_enable` determine which connections are managed by the System Coherency Interface. Connections that are managed by the System Coherency Interface look after themselves for entering and leaving the coherency domain and `debug_force_snoop` has no effect on those ports.

However, for connections that are not managed by the System Coherency Interface, `debug_force_snoop` enables the system upstream of a port to be snooped if the upstream is not in reset and if `acchannelen_rnf` allows it.

Therefore, software does not need to know how to turn snoops in the interconnect on and off. The reset state of the upstream rnf ports is determined by the `rnf_upstream_reset_state[]` signals and must be connected for any port that could go into reset and you want managed by `debug_force_snoop`.

If the connection is managed by SCI or can never go into reset then you need not connect this and the interconnect assumes that the upstream system is not in reset and will always enable the snoops if `ACCHANNELENSX` allows it.

If you fail to connect `rnf_upstream_reset_state[]` correctly, then the upstream system might receive snoop messages while in reset and it will complain that it received a snoop request while it was in reset.

If software overrides the enables then the next change to `rnf_upstream_reset_state[]` or the interconnect reset will overwrite the software values.

Using this operation along with software controlling the enables could confuse the software. This option does not remove the responsibility of the upstream system to clean any shared dirty data from its caches before going into reset.



This parameter is for debug purposes only. Do not use it instead of correctly configuring CMN.

Ports for CMN700

Table 3-896: Ports

Name	Protocol	Type	Description
event_clusters	Signal	Peer	CPU event communication signal from the clusters.
event_downstream_link_signal[32]	Signal	Master	CPU event communication signal to the CMLHub. Event from the CMN towards the Hub NOTE that these are virtual "links" on underlying PCIe hardware bus.
event_upstream_link_signal[32]	Signal	Slave	Event from the Hub towards the CMN
ic_dr_a4s	PVBus	Slave	Interrupt Controller Distributor-to-Remote AXI4Stream port.
ic_rd_a4s	PVBus	Master	Interrupt Controller Remote-to-Distributor AXI4Stream port.
intreqerrns_irq_out	Signal	Master	Interrupt signal
intreqerrs_irq_out	Signal	Master	Interrupt signal
intreqfaultns_irq_out	Signal	Master	Interrupt signal
intreqfaults_irq_out	Signal	Master	Interrupt signal
pvbus_m_cml	PVBus	Master	CML downstream ports
pvbus_m_cml_cfg	PVBus	Master	CML downstream hub configuration port
pvbus_m_cxs[32]	PVBus	Master	CXS downstream ports
pvbus_m_hni[32]	PVBus	Master	HNI downstream ports.
pvbus_m_snf[128]	PVBus	Master	SNF downstream port.
pvbus_s_apb	PVBus	Slave	APB interface port.
pvbus_s_ccg_apb[32]	PVBus	Slave	CCG APB interface port.
pvbus_s_cml	PVBus	Slave	CML upstream ports
pvbus_s_rnf[256]	PVBus	Slave	RNF upstream ports.
pvbus_s_rni[270]	PVBus	Slave	RNI upstream ports. NOTE the upper 150 ports are only used in r2/r3.
reset_in	Signal	Slave	Reset signal.
rnf_sci_s[256]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-F in and out of the coherency domain.
rnf_upstream_reset_in[256]	Signal	Slave	Used by the interconnect to determine the reset state of the RNF manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected.
rni_sci_s[120]	SystemCoherencyInterface	Slave	System Coherency port to move the RN-D in and out of the coherency domain.
rni_upstream_reset_in[270]	Signal	Slave	Used by the interconnect to determine the reset state of the RNI manager to check an aspect of reset sequencing. If debug_force_snoop is used, this port must be connected. NOTE the upper 150 ports are only used in r2/r3.

Name	Protocol	Type	Description
rx_cxs_a4s[32]	PVBus	Slave	Receive channel of A4S packets from a remote CMN.
tx_cxs_a4s[32]	PVBus	Master	Transmit channel of A4S packets to a remote CMN.

Parameters for CMN700

a4s_logicalid

A4S ID mapping of the GIC destination component connected through a CCG port. Specify the CCG NODE ID and the destination A4S Logical ID of the GIC component connected by using a decimal number format like - <CCG_NODEID0>=<A4S_LID0>,<CCG_NODEID1>=<A4S_LID1>. For example for CCG Node ID 54 with A4S ID 12 - 54=12. All of the CCG nodes must be specified. The parameter is only valid when the enable_a4s is also enabled. The default behavior without this parameter is to automatically assign an incrementing A4S ID

Type

string

Default value

acchannelen_rnf

DEPRECATED: Will be removed after FM 11.18. Use 'rnf_sci_enable' instead. For each rnf port, indicates if the port is populated with a snoop responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

acchannelen_rni

DEPRECATED: Will be removed after FM 11.18. Use 'rni_sci_enable' instead. For each rni port, indicates if the port is populated with a dvm responding device or not. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0

cache_state_modelled

Model the cache state.

Type

bool

Default value

0x0

cmn700_tag_cache.metadata_controller.init_value

Initialize metadata memory with this value. If one of `init_values_json` or `init_values_json_file` is specified, this value applies only to any metadata not specified in the JSON.

Type

int

Default value

0xd

cmn700_tag_cache.metadata_controller.init_values_json

A JSON value describing initial metadata values. Mutually exclusive with `init_values_json_file`. The format is as follows: { "regions": [{ "begin": 0x0, "end": 0x10000, "mte_tag": 0xa }, { "begin": 0x20000, "end": 0x50000, "mte_tag": 0xc }] }

Type

string

Default value**cmn700_tag_cache.metadata_controller.init_values_json_file**

Path to a JSON file with initial metadata values. Mutually exclusive with `init_values_json`. The format is as follows: { "regions": [{ "begin": 0x0, "end": 0x10000, "mte_tag": 0xa }, { "begin": 0x20000, "end": 0x50000, "mte_tag": 0xc }] }

Type

string

Default value**cmn700_tag_cache.metadata_controller.is_enabled**

If false, disables the MetaData controller functionality, and makes the component invisible to passing transactions.

Type

bool

Default value

0x0

cmn700_tag_cache.metadata_controller.mte_tag_carveout_json

JSON string that specifies the PA range of the tag carveout regions and the beginning of the PA range for which they provide tag storage. If `pa_regions_with_metadata_storage` defines which regions can have metadata, the tag carveout regions cannot overlap them, and each tagged region must be entirely covered by one of them. The block size must be ≥ 64 bytes and a power of 2, defaulting to 4KiB. The maximum block size supported is 4KiB. The carveout region size must be ≥ 4 KiB and a power of 2, and determines the size of the corresponding tagged region. { "regions": [{ "begin": 0x0, "tag_carveout_region": [0xffffffff0000, 0xffffffff00ff], { "begin": 0x20000, "tag_carveout_region": [0xffffffff0100, 0xffffffff01ff], "block_size": 0x100 }, { "begin": 0x100000, "tag_carveout_region": [0xffffffff0800, 0xffffffff0Bff], "block_size": 0x2000 },] }

Type

string

Default value**cmn700_tag_cache.metadata_controller.mte_tag_carveout_json_file**

Path to a JSON file that specifies the PA range of the tag carveout regions with the same format as mte_tag_carveout_json. Only one of mte_tag_carveout_json and mte_tag_carveout_json_file can be used.

Type

string

Default value**cmn700_tag_cache.metadata_controller.mte_tag_carveout_tag_order**

Order of the tags within the MTE tag carveout blocks. This can be little-endian (same order as the corresponding tagged data) or big-endian (reverse order). The parameter accepts both '-' and '_', so 'little-endian', 'big-endian', 'little_endian' and 'big_endian' are all valid. THIS PARAMETER HAS NO FUNCTIONALITY AT THE MOMENT.

Type

string

Default value

little-endian

cmn700_tag_cache.metadata_controller.pa_regions_with_metadata_storage

Specify the address region where the metadata storage is available for each PAS in a JSON format. If the PAS does not have a region specified, the PAS has metadata storage for all of the space. The regions are defined by begin and end_incl addresses. Example: { "ns": [0xa0000000, 0xa0000fff], "s": [0xb0000000, 0xb0000fff], "rl": [0xc0000000, 0xc0000fff], "rt": [0xd0000000, 0xd0000fff] } ns: non-secure, s: secure, rl: realm, rt: root

Type

string

Default value**debug_force_snoop**

The CMN700 interconnect will normally start with snooping disabled. The parameter rnf_sci_enable and rni_sci_enable determines which connections are managed by the System Coherency Interface. For connections that are managed by the System Coherency Interface, then they will look after themselves for entering and leaving the coherency domain and this parameter has no effect for those ports. However, for connections that are _not_ managed by the System Coherency Interface, this parameter enables the upstream system of a port to be snooped if the upstream is not in reset and if acchannelen_rnf allows it. NOTE: This parameter is for debug purpose only Do not use this parameter instead of correctly configuring CMN. Refer CMN700 documentation page for more info about this parameter in fast models reference manual

Type

bool

Default value

0x0

disable_CML_port

If true the model won't connect the CML port when there are nodes which support the CML feature (ie CXG nodes).

Type

bool

Default value

0x0

enable_a4s

Enables A4S ports for GIC multi-chip routing.

Type

bool

Default value

0x0

enable_logger

Enable PVBUSLoggers for the downstream ports in the CMN model.

Type

bool

Default value

0x0

enable_ras

Enables RAS. There is an impact on performance when RAS is enabled.

Type

bool

Default value

0x0

enable_rnsam_to_hnf_wider_hash

Enable support of wider hash for the RNSAM to HNF communication. If this variable enabled, then bits[47:6] from PA used in hashing function. By default it is [47:12].

Type

bool

Default value

0x0

force_rnsam_internal

Force all RNSAMs to be internal independently of the mesh topology.

Type

bool

Default value

0x1

hnf_mpam_idr_override

Set to override hnf_mpam_idr[31:24] value. Bit[28] is Reserved and is ignored.

Type

int

Default value

0x0

mesh_config_file

Name of a file containing mesh placement of CMN700 components.

Type

string

Default value

/tmp/plgbuild/abs_build/1378878_61223/trunk/work/fastsim/autodocs/../../RegTests/
Interconnect/CMN700/cmn700_genesis_small.yml

periphbase

Value for PERIPHBASE. Bits [27:0] are treated 0

Type

int

Default value

0x20000000

print_cmn_ccix_config

Print information about the CCIX configuration.

Type

bool

Default value

0x0

print_cmn_config

Print the mesh topology and children pointers acquired from the YAML file.

Type

bool

Default value

0x0

register_traces_for_ccg_apb_accesses

Will be removed when enhancement SDDKW-74284 is done. Intended for use with trace plugins. true = registers traces to CCG register accesses through CCG APB interface. false = registers traces to CMN register accesses through all other interfaces (eg RN nodes).

Type

bool

Default value

0x0

revision

Component revision. Currently supports r0p0, r1p0, r2p0, r3p0, r3p3. The version in the mesh_config_file takes priority.

Type

string

Default value

r0p0

rnf_sci_enable

For each rnf port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

rni_sci_enable

For each rni port, indicates if the port is managed by System Coherency Interface for coherency domain entry/exit 1 = Managed by SCI, 0 = Managed by Software. The input value is a string, for example '0xffff' or 'ffff'

Type

string

Default value

0x0

show_banner

Show component banner: 0=>supress entire banner, 1=>suppress config file, 2+=>show full banner.

Type

int

Default value

0x2

skip_cmn_config_check

Skip any topology configuration checks. The maximum number of devices per type not verified.

Type

bool

Default value

0x0

use_yaml_periphbase

Use yaml param CFGM_PERIPHBASE_PARAM to specify periphbase address. If false, model parameter 'periphbase' will be used.

Type

bool

Default value

0x0

yaml_has_node_addresses

Does the top-level YML file describe node-addresses ?

Type

bool

Default value

0x0

3.10.19 DMA350

This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-897: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DMA350

This model has the following Iris instances:

Table 3-898: DMA350 Iris instances

InstanceName	ComponentName
DMA350	DMA350
DMA350.pvbus_m0_bus_master	PVBusMaster
DMA350.pvbus_m1_bus_master	PVBusMaster
DMA350.pvbus_s_bus_slave	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_0	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_1	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_10	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_11	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_12	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_13	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_14	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_15	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_2	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_3	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_4	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_5	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_6	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_7	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_8	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_9	PVBusSlave
DMA350.pvbus_stream_out_bus_master_ch_0	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_1	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_10	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_11	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_12	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_13	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_14	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_15	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_2	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_3	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_4	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_5	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_6	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_7	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_8	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_9	PVBusMaster

This model has the following MTI trace components:

Table 3-899: DMA350 MTI instances

InstanceName	ComponentName
DMA350	DMA350
DMA350.pvbus_m0_bus_master	PVBusMaster
DMA350.pvbus_m1_bus_master	PVBusMaster
DMA350.pvbus_s_bus_slave	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_0	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_1	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_10	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_11	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_12	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_13	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_14	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_15	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_2	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_3	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_4	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_5	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_6	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_7	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_8	PVBusSlave
DMA350.pvbus_stream_in_bus_slave_ch_9	PVBusSlave
DMA350.pvbus_stream_out_bus_master_ch_0	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_1	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_10	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_11	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_12	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_13	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_14	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_15	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_2	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_3	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_4	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_5	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_6	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_7	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_8	PVBusMaster
DMA350.pvbus_stream_out_bus_master_ch_9	PVBusMaster

DMA350 contains the following CADI targets:

- DMA350

About DMA350

This model supports the following functionality:

- 1-8 DMA channels
- 1D memory copy including increments, auto-reload, and command linking
- Interrupt capability for each channel
- 2D memory copy
- 1DWRAP and 2DWRAP support
- Template-based pack and unpack capability
- Security settings per channel
- Trigger input and output ports selectable for each channel
- General Purpose Output (GPO) per channel
- Streaming input and output interfaces per channel
- ADDR_WIDTH and DATA_WIDTH can be 32 bits or 64 bits

Ports for DMA350

Table 3-900: Ports

Name	Protocol	Type	Description
allch_pause_ack_nonsec	Signal	Master	-
allch_pause_ack_sec	Signal	Master	-
allch_pause_req_nonsec	Signal	Slave	Channel pause req/ack for all nonsecure channels
allch_pause_req_sec	Signal	Slave	Channel pause req/ack for all secure channels
allch_stop_ack_nonsec	Signal	Master	-
allch_stop_ack_sec	Signal	Master	-
allch_stop_req_nonsec	Signal	Slave	Channel stop req/ack for all nonsecure channels
allch_stop_req_sec	Signal	Slave	Channel stop req/ack for all secure channels
boot_addr	Value_64	Slave	Address when boot_en is enabled
boot_en	Signal	Slave	Enables channel 0 to load first command after reset from boot_addr
boot_memattr	Value	Slave	Memory attribute setting for the boot_addr
boot_shareattr	Value	Slave	Shareability attribute for the boot_attr
ch_enabled[16]	Signal	Master	Enable status indicator per channel
ch_err[16]	Signal	Master	Error status indicator per channel
ch_nonsec[16]	Signal	Master	Nonsecure status indicator per channel
ch_paused[16]	Signal	Master	Paused status indicator per channel
ch_priv[16]	Signal	Master	Privilege status indicator per channel
ch_stopped[16]	Signal	Master	Stopped status indicator per channel

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Ada DMA clock
gpo_ch[16]	Value	Master	MISC signals General purpose output for channels 0-15 Index refers to the channel
irq_channel[16]	Signal	Master	Channel IRQ Signals
irq_comb_nonsec	Signal	Master	Nonsecure IRQ Signal
irq_comb_nonsec_err	Signal	Master	Nonsecure error IRQ Signal
irq_comb_sec	Signal	Master	Secure IRQ Signal
irq_comb_sec_err	Signal	Master	Secure error IRQ Signal
irq_sec_viol_err	Signal	Master	Security violation IRQ Signal
pvbust_m0	PVBus	Master	AXI5 Master 0 Interface
pvbust_m1	PVBus	Master	AXI5 Master 1 Interface
pvbust_s	PVBus	Slave	APB4 Slave Interface
pvbust_stream_in[16]	PVBus	Slave	AXI-Stream In Interface
pvbust_stream_out[16]	PVBus	Master	AXI-Stream Out Interface
reset_in	Signal	Slave	Ada DMA asynchronous reset
trig_in_ack[32]	Signal	Master	Trigger In Acknowledgement Interface
trig_in_ack_type[32]	Value	Master	-
trig_in_req[32]	Signal	Slave	Trigger In Request Interface
trig_in_req_type[32]	Value	Slave	-
trig_out_ack[32]	Signal	Slave	Trigger Out Acknowledgement Interface
trig_out_req[32]	Signal	Master	Trigger Out Request Interface

Parameters for DMA350

ADDR_WIDTH

Address width of the bus interface

Type

int

Default value

0x20

AXI5_M1_ADDRESS_RANGES

Address ranges for AXI5 M1 interface in the format e.g. [{"begin":0x40000000, "size":0x1000}, {"begin":0x80000000, "size":0x2000}]. Default when not specified uses AXI5 M0 interface

Type

string

Default value

AXI5_M1_PRESENT

Enables an additional master port. When set the m1 master port is present on the top level port list and additional include file can be used with a System Verilog function that defines which address ranges are mapped to the m1 interface.

Type

bool

Default value

0x0

CHID_WIDTH

Width of the configurable channel ID user signal. When set to 0, then the archid and awchid ports are not present on the module.

Type

int

Default value

0x0

CH_0_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_10_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_11_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the

bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_12_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_13_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_14_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_15_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which

can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_1_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_2_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_3_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_4_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_5_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_6_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_7_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_8_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_9_FIFO_DEPTH

Sets the FIFO DEPTH the channels <N> can send for a transaction. N goes from 0 to NUM_CHANNELS-1. This relates to the maximum burst size a channel can support which can be calculated as $\min(16, (\text{FIFO_DEPTH}+1)/2)$. This setting needs to be aligned with the bandwidth requirements of the selected channel. This parameter highly affects the area of the design.

Type

int

Default value

0x2

CH_EXT_FEAT_EN

Enabling the extended feature set for each channel. The extension contains 2D, WRAP, TMPLT features. Default value enables it for the number of channels.

Type

int

Default value

0x1

CH_GPO_EN**Type**

int

Default value

0x1

CH_GPO_MASK

A bitmask for enabling the GPO port for each channel. The width of the bitmask is NUM_CHANNELS-1. When bit n is set to 1 then the GPO is enabled for channel n and the gpo_ch_n[GPO_WIDTH-1:0] port appears on the module.

Type

int

Default value

0x0

CH_STREAM_EN**Type**

int

Default value

0x1

CH_STREAM_MASK

A bitmask for enabling the stream interfaces for each channel. The width of the bitmask is NUM_CHANNELS-1. When bit n is set to 1 then the stream interfaces are enabled for channel n and the relevant ports appears on the module. NOTE: When streaming interface is enabled the actual FIFO size of the channel will be the double of CH_<N>_FIFO_DEPTH

Type

int

Default value

0x0

DATA_WIDTH

Data width of the bus interface

Type

int

Default value

0x40

DISABLE_DEVICE

Disable device and ignore all interfaces

Type

bool

Default value

0x0

DUMP_CONFIG

Display DMA-350 DMAC parameters

Type

bool

Default value

0x0

GPO_WIDTH

Width of GPO output for every channel. When multiple channels have GPOs then the width must be set to the maximum number of GPOs a channel can have, and unused GPO ports need to be left unconnected. When all bits of CH_GPO_MASK is 0, this parameter is not relevant.

Type

int

Default value

0x1

NUM_CHANNELS

Number of configurable DMA channels.

Type

int

Default value

0x2

NUM_TRIGGER_IN

Number of trigger input ports.

Type

int

Default value

0x2

NUM_TRIGGER_OUT

Number of trigger output ports.

Type

int

Default value

0x2

SECEXT_PRESENT

Enables Trustzone security support.

Type

bool

Default value

0x1

3.10.20 DMC500

ARM Dynamic Memory Controller(DMC500). This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-901: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DMC500

This model has the following Iris instances:

Table 3-902: DMC500 Iris instances

InstanceName	ComponentName
DMC500	DMC500
DMC500.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-903: DMC500 MTI instances

InstanceName	ComponentName
DMC500	DMC-500
DMC500.busslave	PVBusSlave

DMC500 contains the following CADI targets:

- DMC500

About DMC500

A platform can have multiple instances of this component. For example:

```
//LISA instantiation
composition
{
    // Memory controllers
    dmc0      : DMC500("default_region_attributes"=dmc_default_region_attributes,
                      "default_region_id_access"=dmc_default_region_id_access,
                      "passthrough_debug_access"=true);
    dmc1      : DMC500("default_region_attributes"=dmc_default_region_attributes,
                      "default_region_id_access"=dmc_default_region_id_access,
                      "passthrough_debug_access"=true);
}
```

Differences between the model and the RTL

The model has the following limitations:

- It does not support address striping.

- It works with linear addresses and not in rank,bank,row,column form.
- It does not include any mechanism for error injection or detection.
- Scrubbing functionality is only provided from the interface point of view.
- It does not implement direct read or write commands.
- It does not implement any performance counters.
- All OR'd interrupt signals are missing from this release of the model. Users can connect the failed access interrupt as a substitute.
- The model combines separate failed access interrupts for system interfaces 1 and 2 into a single failed access interrupt.
- DMC-500 has three separate reset signals whereas this model has a single reset signal which supports the combined assertion of three resets. This model does not support separate reset signals.

Ports for DMC500

Table 3-904: Ports

Name	Protocol	Type	Description
apb_pvbus_s	PVBus	Slave	Programmers interface to program and control the DMC-500.
failed_access_interrupt_signal	Signal	Master	The DMC has detected a system request that has failed a permissions check and a previously detected assertion was not cleared.
filter_pvbus_m	PVBus	Master	DMC master port from System Interface 0 to memory.
filter_pvbus_s	PVBus	Slave	System interface 0. Generally, Non-coherent Interface.
reset_signal	Signal	Slave	DMC reset.
si1_filter_pvbus_m	PVBus	Master	DMC master port from System Interface 1 to memory.
si1_filter_pvbus_s	PVBus	Slave	System interface 1. Generally, Coherent Interface.

Parameters for DMC500

default_region_attributes

Default Region Secure attributes. Only bits 31,30 set Secure RD/WR enable.

Type

int

Default value

0x1

default_region_id_access

Default Region NSAID permissions. Bits 31-16 set non-secure WR enable and bits 15-0 set non-secure RD enable.

Type

int

Default value

0x0

passthrough_debug_access

Always allow debug access to memory.

Type

bool

Default value

0x0

3.10.21 DMC520

ARM Dynamic Memory Controller(DMC520). This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-905: IP revisions support

Revision	Quality level
r1p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DMC520

This model has the following Iris instances:

Table 3-906: DMC520 Iris instances

InstanceName	ComponentName
DMC520	DMC520
DMC520.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-907: DMC520 MTI instances

InstanceName	ComponentName
DMC520	DMC-520
DMC520.busslave	PVBusSlave

DMC520 contains the following CADI targets:

- [DMC520](#)

About DMC520

A platform can have multiple instances of this component. For example:

```
//LISA instantiation
composition
{
    // Memory controllers
```

```

dmc520_0      : DMC520 ("passthrough_debug_access"=true);
dmc520_1      : DMC520 ("passthrough_debug_access"=true);
}

```

Limitations:

- The model does not support address striping.
- It works with linear addresses and not in rank,bank,row,column form.
- It does not include any mechanism for error injection or detection.
- Scrubbing functionality is only provided from the interface point of view.
- It does not implement direct read or write commands.
- It does not implement any performance counters.

Differences between the model and the RTL

The DMC520 and DMC620 models have different interfaces to those in the hardware due to the level of abstraction of memory in Fast Models. These are the differences:

- Like the hardware, the model has a slave port for configuring register accesses, `apb_pvbus_s`, and an AXI interface for incoming memory transactions that are attempting to access memory that is managed by the DMC.
- The hardware component translates incoming transactions on the AXI interface to a format that is conducive to accessing DRAM chips. The model performs TrustZone® access control and models the DMC readiness state, but does not translate the transactions. If allowed, the model forwards incoming transactions to be handled by a memory storage handling component that works at the transaction level.

Ports for DMC520

Table 3-908: Ports

Name	Protocol	Type	Description
<code>all_or_interrupt_signal</code>	Signal	Master	A combined interrupt that is the logical OR of the other interrupts.
<code>apb_pvbus_s</code>	PVBus	Slave	Programmers interface to program and control the DMC-520.
<code>arch_fsm_interrupt_signal</code>	Signal	Master	The DMC has detected a change in the architectural state.
<code>failed_access_interrupt_signal</code>	Signal	Master	The DMC has detected a system request that has failed a permissions check and a previously detected assertion was not cleared.
<code>filter_pvbus_m</code>	PVBus	Master	DMC master port to memory.
<code>filter_pvbus_s</code>	PVBus	Slave	System interface.
<code>reset_signal</code>	Signal	Slave	DMC reset.
<code>scrub_event_in[8]</code>	Signal	Slave	Scrub event n trigger.
<code>scrub_event_out[8]</code>	Signal	Master	Scrub event n triggered.

Parameters for DMC520

`override_default_config`

Override default block-all behavior of DMC. Allow access to memory.

Type

bool

Default value

0x0

passthrough_debug_access

Always allow debug access to memory.

Type

bool

Default value

0x0

3.10.22 DMC620

ARM Dynamic Memory Controller(DMC620). This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-909: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DMC620

This model has the following Iris instances:

Table 3-910: DMC620 Iris instances

InstanceName	ComponentName
DMC620	DMC620
DMC620.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-911: DMC620 MTI instances

InstanceName	ComponentName
DMC620	DMC-620
DMC620.busslave	PVBusSlave

DMC620 contains the following CADI targets:

- DMC620

About DMC620

The model has the following limitations:

- It does not support address striping.
- It works with linear addresses and not in rank, bank, row, column form.
- It includes error injection and detection mechanisms and syndrome registers support only for RAS error types 4 (ECC single-bit SRAM error) and 5 (ECC double-bit SRAM error).
- Scrubbing functionality is not provided.
- It does not implement direct read or write commands.
- It does not implement any performance counters.

Differences between the model and the RTL

The DMC520 and DMC620 models have different interfaces to those in the hardware due to the level of abstraction of memory in Fast Models. These are the differences:

- Like the hardware, the model has a slave port for configuring register accesses, `apb_pvbus_s`, and an AXI interface for incoming memory transactions that are attempting to access memory that is managed by the DMC.
- The hardware component translates incoming transactions on the AXI interface to a format that is conducive to accessing DRAM chips. The model performs TrustZone® access control and models the DMC readiness state, but does not translate the transactions. If allowed, the model forwards incoming transactions to be handled by a memory storage handling component that works at the transaction level.

Ports for DMC620

Table 3-912: Ports

Name	Protocol	Type	Description
<code>all_or_interrupt_signal</code>	Signal	Master	A combined interrupt that is the logical OR of the other interrupts.
<code>apb_pvbus_s</code>	PVBus	Slave	Programmers interface to program and control the DMC-620.
<code>arch_fsm_interrupt_signal</code>	Signal	Master	The DMC has detected a change in the architectural state.
<code>failed_access_interrupt_signal</code>	Signal	Master	The DMC has detected a system request that has failed a permissions check and a previously detected assertion was not cleared.
<code>filter_pvbus_m</code>	PVBus	Master	DMC master port to memory.
<code>filter_pvbus_s</code>	PVBus	Slave	System interface.
<code>interrupt_cfh_master</code>	Signal	Master	The DMC has detected and corrected a single bit error on the RAM access.
<code>interrupt_combined_oflow_master</code>	Signal	Master	The DMC has detected a counter overflow.
<code>interrupt_fh_master</code>	Signal	Master	The DMC has detected a double bit error on the RAM access.
<code>reset_signal</code>	Signal	Slave	DMC reset.

Parameters for DMC620

override_default_config

Override default block-all behavior of DMC. Allow access to memory.

Type

bool

Default value

0x0

passthrough_debug_access

Always allow debug access to memory.

Type

bool

Default value

0x0

3.10.23 DMC_400

ARM PrimeCell Dynamic Memory Controller(DMC400). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-913: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for DMC_400

This model has the following Iris instances:

Table 3-914: DMC_400 Iris instances

InstanceName	ComponentName
DMC_400	DMC_400
DMC_400.apb_slave	PVBusSlave
DMC_400.ex_mon0	PVBusMapper
DMC_400.ex_mon1	PVBusMapper
DMC_400.ex_mon2	PVBusMapper
DMC_400.ex_mon3	PVBusMapper

This model has the following MTI trace components:

Table 3-915: DMC_400 MTI instances

InstanceName	ComponentName
DMC_400.apb_slave	PVBusSlave
DMC_400.ex_mon0	PVBusMapper
DMC_400.ex_mon1	PVBusMapper
DMC_400.ex_mon2	PVBusMapper
DMC_400.ex_mon3	PVBusMapper

DMC_400 contains the following CADI targets:

- DMC_400

About DMC_400

The configuration of this model by setting the registers does not generally affect accesses to main memory.

This component has no timing information, so changing the values of the timing registers has no effect on behavior. The memory models do not attach to the component, and error checking does not update registers because the model does not include the possibility of errors.

Ports for DMC_400

Table 3-916: Ports

Name	Protocol	Type	Description
apb_interface	PVBus	Slave	Slave bus interface for register access.
axi_if_in[4]	PVBus	Slave	Slave bus for connecting to bus decoder.
axi_if_out[4]	PVBus	Master	Master to connect to DRAM.
clr_ex_mon	Signal	Master	Indicates when global monitors state is cleared.
user_status_ext	Value	Slave	Allow user status to be set from outside.

Parameters for DMC_400

ECC_SUPPORT

Does the controller support ECC?

Type

bool

Default value

0x1

IF_CHIP0

Set this parameter to 0 if memory is connected

Type

int

Default value`0xffffffffffffffffffff`**IF_CHIP1**

Set this parameter to 0 if memory is connected

Type`int`**Default value**`0xffffffffffffffffffff`**IF_CHIP2**

Set this parameter to 0 if memory is connected

Type`int`**Default value**`0xffffffffffffffffffff`**IF_CHIP3**

Set this parameter to 0 if memory is connected

Type`int`**Default value**`0xffffffffffffffffffff`**MEMORY_WIDTH**

Valid widths are 16, 32 or 64 bits

Type`int`**Default value**`0x20`**diagnostics**

Diagnostics

Type`int`**Default value**`0x0`**revision_string**

Revision

Type
string

Default value
rOp1

3.10.24 EthosU55

Arm Ethos-U55 microNPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-917: IP revisions support

Revision	Quality level
r2p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for EthosU55

This model has the following Iris instances:

Table 3-918: EthosU55 Iris instances

InstanceName	ComponentName
EthosU55	EthosU55
EthosU55.pvbusmaster0	PVBusMaster
EthosU55.pvbusmaster1	PVBusMaster
EthosU55.pvbusslave	PVBusSlave

This model has the following MTI trace components:

Table 3-919: EthosU55 MTI instances

InstanceName	ComponentName
EthosU55.pvbusmaster0	PVBusMaster
EthosU55.pvbusmaster1	PVBusMaster
EthosU55.pvbusslave	PVBusSlave

EthosU55 contains the following CADI targets:

- EthosU55

About EthosU55

The EthosU55 model has the following parameters:

diagnostics

Enables additional information messages from the EthosU55 component about the NPU configuration and the PORPL/PORSL port values. Use any nonzero integer to enable all messages. The default value of 0 disables them.

extra_args

Reserved for future use. We recommend using this parameter only with customized instructions from [Arm Technical Support](#).

num_macs

Controls the number of 8x8 MACs performed per cycle. The valid values are 32, 64, 128, or 256. The default value is 128.



- The EthosU55 model does not expose its registers through CADI or Iris.
- The `resetrn_in` signal is active-LOW.

Ports for EthosU55**Table 3-920: Ports**

Name	Protocol	Type	Description
<code>clk_in</code>	ClockSignal	Slave	NPU clock signal
<code>irq_out</code>	Signal	Master	Sends interrupt requests to the external host application processor
<code>popl_in</code>	Signal	Slave	The Power-On-Reset Privilege Level (PORPL). LOW means User level. HIGH means Privileged level
<code>posl_in</code>	Signal	Slave	The Power-On-Reset Security Level (PORSL). LOW means Secure. HIGH means Non-secure.
<code>pvbuss_m0</code>	PVBus	Master	Port 0 for NPU to access external memory
<code>pvbuss_m1</code>	PVBus	Master	Port 1 for NPU to access external memory
<code>pvbuss_s</code>	PVBus	Slave	Port to access NPU control registers
<code>resetrn_in</code>	Signal	Slave	NPU reset signal

Parameters for EthosU55**diagnostics**

Enable diagnostic messages

Type

int

Default value

0x0

extra_args

Reserved for future use. We recommend using this parameter only with customized instructions from Arm Technical Support (support-esl@arm.com).

Type

string

Default value**num_macs**

Number of 8x8 MACs performed per cycle

Type

int

Default value

0x80

3.10.25 EthosU65

Arm Ethos-U65 microNPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-921: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for EthosU65

This model has the following Iris instances:

Table 3-922: EthosU65 Iris instances

InstanceName	ComponentName
EthosU65	EthosU65
EthosU65.pvbusmaster0	PVBusMaster
EthosU65.pvbusmaster1	PVBusMaster
EthosU65.pvbusslave	PVBusSlave

This model has the following MTI trace components:

Table 3-923: EthosU65 MTI instances

InstanceName	ComponentName
EthosU65.pvbusmaster0	PVBusMaster
EthosU65.pvbusmaster1	PVBusMaster
EthosU65.pvbusslave	PVBusSlave

EthosU65 contains the following CADI targets:

- EthosU65

About EthosU65

The EthosU65 model has the following parameters:

diagnostics

Enables additional information messages from the EthosU65 component about the NPU configuration and the PORPL/PORSL port values. Use any nonzero integer to enable all messages. The default value of 0 disables them.

extra_args

Reserved for future use. We recommend using this parameter only with customized instructions from [Arm Technical Support](#).

num_macs

Controls the number of 8x8 MACs performed per cycle. The valid values are 256 and 512. The default value is 256.



- The EthosU65 model does not expose its registers through CADI or Iris.
- The `resetrn_in` signal is active-LOW.

Ports for EthosU65

Table 3-924: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	NPU clock signal
irq_out	Signal	Master	Sends interrupt requests to the external host application processor
popl_in	Signal	Slave	The Power-On-Reset Privilege Level (PORPL). LOW means User level. HIGH means Privileged level
posl_in	Signal	Slave	The Power-On-Reset Security Level (PORSL). LOW means Secure. HIGH means Non-secure.
pvbus_m0	PVBus	Master	Port 0 for NPU to access external memory
pvbus_m1	PVBus	Master	Port 1 for NPU to access external memory
pvbus_s	PVBus	Slave	Port to access NPU control registers
resetrn_in	Signal	Slave	NPU reset signal

Parameters for EthosU65

diagnostics

Enable diagnostic messages

Type

int

Default value

0x0

extra_args

Reserved for future use. We recommend using this parameter only with customized instructions from [Arm Technical Support \(support-esl@arm.com\)](mailto:support-esl@arm.com).

Type

string

Default value

num_macs

Number of 8x8 MACs performed per cycle

Type

int

Default value

0x100

3.10.26 EthosU85

Arm EthosU85 microNPU. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-925: IP revisions support

Revision	Quality level
r0p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for EthosU85

This model has the following Iris instances:

Table 3-926: EthosU85 Iris instances

InstanceName	ComponentName
EthosU85	EthosU85
EthosU85.pvbusmaster0	PVBusMaster
EthosU85.pvbusmaster1	PVBusMaster
EthosU85.pvbusslave	PVBusSlave

This model has the following MTI trace components:

Table 3-927: EthosU85 MTI instances

InstanceName	ComponentName
EthosU85.pvbusmaster0	PVBusMaster
EthosU85.pvbusmaster1	PVBusMaster
EthosU85.pvbusslave	PVBusSlave

EthosU85 contains the following CADI targets:

- EthosU85

Ports for EthosU85

Table 3-928: Ports

Name	Protocol	Type	Description
cfgextcap_in	Value	Slave	The configuration of capabilities for DRAM AXI ports (32 bits). Sampled with soft and hard reset.
cfgexthash0_in	Value_64	Slave	The configuration of hash function for selecting among EXT ports (40 bits). Used to set the hash for AXI DRAM ports 0 and 1 if they are present. Sampled with soft and hard reset.
cfgsramcap_in	Value	Slave	The configuration of capabilities for SRAM AXI ports (32 bits). Sampled with soft and hard reset.
cfgsramhash0_in	Value_64	Slave	The configuration of hash function for selecting among SRAM ports (40 bits). Used to set the hash for AXI SRAM ports 0 and 1. Sampled with soft and hard reset.
cfgsramhash1_in	Value_64	Slave	The configuration of hash function for selecting among SRAM ports (40 bits). Used to set the hash for AXI SRAM ports 2 and 3 if they are present. Sampled with soft and hard reset.
clk_in	ClockSignal	Slave	NPU clock signal
irq_out	Signal	Master	Sends interrupt requests to the external host application processor, level triggered when HIGH.
popl_in	Signal	Slave	The Power-On-Reset Privilege Level (PORPL). LOW means User level. HIGH means Privileged level
posl_in	Signal	Slave	The Power-On-Reset Security Level (PORSL). LOW means Secure. HIGH means Non-secure.
pvbush_m0	PVBus	Master	Port 0 for NPU to access external memory
pvbush_m1	PVBus	Master	Port 1 for NPU to access external memory
pvbush_s	PVBus	Slave	Port to access NPU control registers
reseth_in	Signal	Slave	NPU reset signal (active-LOW)

Parameters for EthosU85

diagnostics

Enable diagnostic messages

Type

int

Default value

0x0

extra_args

Reserved for future use. We recommend using this parameter only with customized instructions from Arm Technical Support (support-esl@arm.com).

Type

string

Default value

num_macs

Number of 8x8 MACs performed per cycle

Type

int

Default value

0x80

3.10.27 Firewall

Firewall IP. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-929: IP revisions support

Revision	Quality level
rOp0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for Firewall

This model has the following Iris instances:

Table 3-930: Firewall Iris instances

InstanceName	ComponentName
Firewall	Firewall
Firewall.BusLogger0	PVBusLogger
Firewall.BusLogger0.mapper	PVBusMapper
Firewall.BusLogger1	PVBusLogger
Firewall.BusLogger1.mapper	PVBusMapper
Firewall.BusLogger10	PVBusLogger
Firewall.BusLogger10.mapper	PVBusMapper
Firewall.BusLogger11	PVBusLogger
Firewall.BusLogger11.mapper	PVBusMapper
Firewall.BusLogger12	PVBusLogger
Firewall.BusLogger12.mapper	PVBusMapper
Firewall.BusLogger13	PVBusLogger
Firewall.BusLogger13.mapper	PVBusMapper
Firewall.BusLogger14	PVBusLogger
Firewall.BusLogger14.mapper	PVBusMapper
Firewall.BusLogger15	PVBusLogger
Firewall.BusLogger15.mapper	PVBusMapper

InstanceName	ComponentName
Firewall.BusLogger16	PVBusLogger
Firewall.BusLogger16.mapper	PVBusMapper
Firewall.BusLogger17	PVBusLogger
Firewall.BusLogger17.mapper	PVBusMapper
Firewall.BusLogger18	PVBusLogger
Firewall.BusLogger18.mapper	PVBusMapper
Firewall.BusLogger19	PVBusLogger
Firewall.BusLogger19.mapper	PVBusMapper
Firewall.BusLogger2	PVBusLogger
Firewall.BusLogger2.mapper	PVBusMapper
Firewall.BusLogger20	PVBusLogger
Firewall.BusLogger20.mapper	PVBusMapper
Firewall.BusLogger21	PVBusLogger
Firewall.BusLogger21.mapper	PVBusMapper
Firewall.BusLogger22	PVBusLogger
Firewall.BusLogger22.mapper	PVBusMapper
Firewall.BusLogger23	PVBusLogger
Firewall.BusLogger23.mapper	PVBusMapper
Firewall.BusLogger24	PVBusLogger
Firewall.BusLogger24.mapper	PVBusMapper
Firewall.BusLogger25	PVBusLogger
Firewall.BusLogger25.mapper	PVBusMapper
Firewall.BusLogger26	PVBusLogger
Firewall.BusLogger26.mapper	PVBusMapper
Firewall.BusLogger27	PVBusLogger
Firewall.BusLogger27.mapper	PVBusMapper
Firewall.BusLogger28	PVBusLogger
Firewall.BusLogger28.mapper	PVBusMapper
Firewall.BusLogger29	PVBusLogger
Firewall.BusLogger29.mapper	PVBusMapper
Firewall.BusLogger3	PVBusLogger
Firewall.BusLogger3.mapper	PVBusMapper
Firewall.BusLogger30	PVBusLogger
Firewall.BusLogger30.mapper	PVBusMapper
Firewall.BusLogger31	PVBusLogger
Firewall.BusLogger31.mapper	PVBusMapper
Firewall.BusLogger4	PVBusLogger
Firewall.BusLogger4.mapper	PVBusMapper
Firewall.BusLogger5	PVBusLogger

InstanceName	ComponentName
Firewall.BusLogger5.mapper	PVBusMapper
Firewall.BusLogger6	PVBusLogger
Firewall.BusLogger6.mapper	PVBusMapper
Firewall.BusLogger7	PVBusLogger
Firewall.BusLogger7.mapper	PVBusMapper
Firewall.BusLogger8	PVBusLogger
Firewall.BusLogger8.mapper	PVBusMapper
Firewall.BusLogger9	PVBusLogger
Firewall.BusLogger9.mapper	PVBusMapper
Firewall.BusMapper0	PVBusMapper
Firewall.BusMapper1	PVBusMapper
Firewall.BusMapper10	PVBusMapper
Firewall.BusMapper11	PVBusMapper
Firewall.BusMapper12	PVBusMapper
Firewall.BusMapper13	PVBusMapper
Firewall.BusMapper14	PVBusMapper
Firewall.BusMapper15	PVBusMapper
Firewall.BusMapper16	PVBusMapper
Firewall.BusMapper17	PVBusMapper
Firewall.BusMapper18	PVBusMapper
Firewall.BusMapper19	PVBusMapper
Firewall.BusMapper2	PVBusMapper
Firewall.BusMapper20	PVBusMapper
Firewall.BusMapper21	PVBusMapper
Firewall.BusMapper22	PVBusMapper
Firewall.BusMapper23	PVBusMapper
Firewall.BusMapper24	PVBusMapper
Firewall.BusMapper25	PVBusMapper
Firewall.BusMapper26	PVBusMapper
Firewall.BusMapper27	PVBusMapper
Firewall.BusMapper28	PVBusMapper
Firewall.BusMapper29	PVBusMapper
Firewall.BusMapper3	PVBusMapper
Firewall.BusMapper30	PVBusMapper
Firewall.BusMapper31	PVBusMapper
Firewall.BusMapper4	PVBusMapper
Firewall.BusMapper5	PVBusMapper
Firewall.BusMapper6	PVBusMapper
Firewall.BusMapper7	PVBusMapper

InstanceName	ComponentName
Firewall.BusMapper8	PVBusMapper
Firewall.BusMapper9	PVBusMapper
Firewall.bus_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-931: Firewall MTI instances

InstanceName	ComponentName
Firewall	Firewall
Firewall.BusLogger0	PVBusLogger
Firewall.BusLogger0.mapper	PVBusMapper
Firewall.BusLogger1	PVBusLogger
Firewall.BusLogger1.mapper	PVBusMapper
Firewall.BusLogger10	PVBusLogger
Firewall.BusLogger10.mapper	PVBusMapper
Firewall.BusLogger11	PVBusLogger
Firewall.BusLogger11.mapper	PVBusMapper
Firewall.BusLogger12	PVBusLogger
Firewall.BusLogger12.mapper	PVBusMapper
Firewall.BusLogger13	PVBusLogger
Firewall.BusLogger13.mapper	PVBusMapper
Firewall.BusLogger14	PVBusLogger
Firewall.BusLogger14.mapper	PVBusMapper
Firewall.BusLogger15	PVBusLogger
Firewall.BusLogger15.mapper	PVBusMapper
Firewall.BusLogger16	PVBusLogger
Firewall.BusLogger16.mapper	PVBusMapper
Firewall.BusLogger17	PVBusLogger
Firewall.BusLogger17.mapper	PVBusMapper
Firewall.BusLogger18	PVBusLogger
Firewall.BusLogger18.mapper	PVBusMapper
Firewall.BusLogger19	PVBusLogger
Firewall.BusLogger19.mapper	PVBusMapper
Firewall.BusLogger2	PVBusLogger
Firewall.BusLogger2.mapper	PVBusMapper
Firewall.BusLogger20	PVBusLogger
Firewall.BusLogger20.mapper	PVBusMapper
Firewall.BusLogger21	PVBusLogger
Firewall.BusLogger21.mapper	PVBusMapper
Firewall.BusLogger22	PVBusLogger

InstanceName	ComponentName
Firewall.BusLogger22.mapper	PVBusMapper
Firewall.BusLogger23	PVBusLogger
Firewall.BusLogger23.mapper	PVBusMapper
Firewall.BusLogger24	PVBusLogger
Firewall.BusLogger24.mapper	PVBusMapper
Firewall.BusLogger25	PVBusLogger
Firewall.BusLogger25.mapper	PVBusMapper
Firewall.BusLogger26	PVBusLogger
Firewall.BusLogger26.mapper	PVBusMapper
Firewall.BusLogger27	PVBusLogger
Firewall.BusLogger27.mapper	PVBusMapper
Firewall.BusLogger28	PVBusLogger
Firewall.BusLogger28.mapper	PVBusMapper
Firewall.BusLogger29	PVBusLogger
Firewall.BusLogger29.mapper	PVBusMapper
Firewall.BusLogger3	PVBusLogger
Firewall.BusLogger3.mapper	PVBusMapper
Firewall.BusLogger30	PVBusLogger
Firewall.BusLogger30.mapper	PVBusMapper
Firewall.BusLogger31	PVBusLogger
Firewall.BusLogger31.mapper	PVBusMapper
Firewall.BusLogger4	PVBusLogger
Firewall.BusLogger4.mapper	PVBusMapper
Firewall.BusLogger5	PVBusLogger
Firewall.BusLogger5.mapper	PVBusMapper
Firewall.BusLogger6	PVBusLogger
Firewall.BusLogger6.mapper	PVBusMapper
Firewall.BusLogger7	PVBusLogger
Firewall.BusLogger7.mapper	PVBusMapper
Firewall.BusLogger8	PVBusLogger
Firewall.BusLogger8.mapper	PVBusMapper
Firewall.BusLogger9	PVBusLogger
Firewall.BusLogger9.mapper	PVBusMapper
Firewall.BusMapper0	PVBusMapper
Firewall.BusMapper1	PVBusMapper
Firewall.BusMapper10	PVBusMapper
Firewall.BusMapper11	PVBusMapper
Firewall.BusMapper12	PVBusMapper
Firewall.BusMapper13	PVBusMapper

InstanceName	ComponentName
Firewall.BusMapper14	PVBusMapper
Firewall.BusMapper15	PVBusMapper
Firewall.BusMapper16	PVBusMapper
Firewall.BusMapper17	PVBusMapper
Firewall.BusMapper18	PVBusMapper
Firewall.BusMapper19	PVBusMapper
Firewall.BusMapper2	PVBusMapper
Firewall.BusMapper20	PVBusMapper
Firewall.BusMapper21	PVBusMapper
Firewall.BusMapper22	PVBusMapper
Firewall.BusMapper23	PVBusMapper
Firewall.BusMapper24	PVBusMapper
Firewall.BusMapper25	PVBusMapper
Firewall.BusMapper26	PVBusMapper
Firewall.BusMapper27	PVBusMapper
Firewall.BusMapper28	PVBusMapper
Firewall.BusMapper29	PVBusMapper
Firewall.BusMapper3	PVBusMapper
Firewall.BusMapper30	PVBusMapper
Firewall.BusMapper31	PVBusMapper
Firewall.BusMapper4	PVBusMapper
Firewall.BusMapper5	PVBusMapper
Firewall.BusMapper6	PVBusMapper
Firewall.BusMapper7	PVBusMapper
Firewall.BusMapper8	PVBusMapper
Firewall.BusMapper9	PVBusMapper
Firewall.bus_slave	PVBusSlave

Firewall contains the following CADI targets:

- Firewall

Ports for Firewall

Table 3-932: Ports

Name	Protocol	Type	Description
irq_signal	Signal	Master	-
irq_signal_tamper	Signal	Master	-
lockdown	Signal	Slave	-
pvbus_component_m[31]	PVBus	Master	-
pvbus_component_s[31]	PVBus	Slave	-

GIC500 contains the following CADI targets:

- GIC500

About GIC500

This is a single-component implementation of the GICv3 architecture with support for 256 cores. You can configure the model to support a maximum of 32 clusters with 8 cores per cluster. Use it with an Arm®v8-A core to deliver interrupts. It supports a single Interrupt Translation Service for message-based interrupts. It supports the architectural features, but does not support the **IMPLEMENTATION DEFINED** features.

To use the GIC500 component, you must configure some parameters. For example:

```
gic500: GIC500(
    "num_clusters" = 2,
    "cpus_per_cluster_0" = 4,
    "cpus_per_cluster_1" = 4,
    "reg-base" = 0x2c200000,
    "SPI-count" = 256
);
```



Note

- To print to `stderr` the memory map of any GICv3 or later models that are included in the platform, set the `FASTSIM_GIC_MEMORY_MAP` environment variable to 1.
- For writes to `GITS_TRANSLATE64R`, the MSIRewriter component can be used to create the correct data transactions based on the device ID. For more details about `GITS_TRANSLATE64R`, see the [3.3.4 MSIRewriter](#) on page 139 component.

MasterID, ExtendedID, and UserFlags

This table shows how this model encodes information in the `MasterID`, `ExtendedID`, and `UserFlags` bus attributes:

Table 3-936: `pvbus_s` port

PVBus attribute	Bits used	Property encoded	Notes
MasterID	Bits[31:0]	Master ID that invoked the register access	-
ExtendedID	-	-	Not used.
UserFlags	-	-	Not used.



Note

The `pvbus_m` port does not use `MasterID`, `ExtendedID`, or `UserFlags`.

Ports for GIC500

Table 3-937: Ports

Name	Protocol	Type	Description
cfgsdisable	Signal	Slave	Disable some SPIs signal.
cpu_active_0[8]	Signal	Slave	cpu_active pins of cluster 0.
cpu_active_1[8]	Signal	Slave	cpu_active pins of cluster 1.
cpu_active_10[8]	Signal	Slave	cpu_active pins of cluster 10.
cpu_active_11[8]	Signal	Slave	cpu_active pins of cluster 11.
cpu_active_12[8]	Signal	Slave	cpu_active pins of cluster 12.
cpu_active_13[8]	Signal	Slave	cpu_active pins of cluster 13.
cpu_active_14[8]	Signal	Slave	cpu_active pins of cluster 14.
cpu_active_15[8]	Signal	Slave	cpu_active pins of cluster 15.
cpu_active_16[8]	Signal	Slave	cpu_active pins of cluster 16.
cpu_active_17[8]	Signal	Slave	cpu_active pins of cluster 17.
cpu_active_18[8]	Signal	Slave	cpu_active pins of cluster 18.
cpu_active_19[8]	Signal	Slave	cpu_active pins of cluster 19.
cpu_active_2[8]	Signal	Slave	cpu_active pins of cluster 2.
cpu_active_20[8]	Signal	Slave	cpu_active pins of cluster 20.
cpu_active_21[8]	Signal	Slave	cpu_active pins of cluster 21.
cpu_active_22[8]	Signal	Slave	cpu_active pins of cluster 22.
cpu_active_23[8]	Signal	Slave	cpu_active pins of cluster 23.
cpu_active_24[8]	Signal	Slave	cpu_active pins of cluster 24.
cpu_active_25[8]	Signal	Slave	cpu_active pins of cluster 25.
cpu_active_26[8]	Signal	Slave	cpu_active pins of cluster 26.
cpu_active_27[8]	Signal	Slave	cpu_active pins of cluster 27.
cpu_active_28[8]	Signal	Slave	cpu_active pins of cluster 28.
cpu_active_29[8]	Signal	Slave	cpu_active pins of cluster 29.
cpu_active_3[8]	Signal	Slave	cpu_active pins of cluster 3.
cpu_active_30[8]	Signal	Slave	cpu_active pins of cluster 30.
cpu_active_31[8]	Signal	Slave	cpu_active pins of cluster 31.
cpu_active_4[8]	Signal	Slave	cpu_active pins of cluster 4.
cpu_active_5[8]	Signal	Slave	cpu_active pins of cluster 5.
cpu_active_6[8]	Signal	Slave	cpu_active pins of cluster 6.
cpu_active_7[8]	Signal	Slave	cpu_active pins of cluster 7.
cpu_active_8[8]	Signal	Slave	cpu_active pins of cluster 8.
cpu_active_9[8]	Signal	Slave	cpu_active pins of cluster 9.
po_reset	Signal	Slave	Power on reset.
ppi16_in_0[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 0.
ppi16_in_1[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 1.
ppi16_in_10[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 10.

Name	Protocol	Type	Description
ppi16_in_11[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 11.
ppi16_in_12[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 12.
ppi16_in_13[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 13.
ppi16_in_14[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 14.
ppi16_in_15[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 15.
ppi16_in_16[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 16.
ppi16_in_17[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 17.
ppi16_in_18[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 18.
ppi16_in_19[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 19.
ppi16_in_2[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 2.
ppi16_in_20[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 20.
ppi16_in_21[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 21.
ppi16_in_22[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 22.
ppi16_in_23[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 23.
ppi16_in_24[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 24.
ppi16_in_25[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 25.
ppi16_in_26[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 26.
ppi16_in_27[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 27.
ppi16_in_28[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 28.
ppi16_in_29[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 29.
ppi16_in_3[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 3.
ppi16_in_30[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 30.
ppi16_in_31[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 31.
ppi16_in_4[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 4.
ppi16_in_5[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 5.
ppi16_in_6[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 6.
ppi16_in_7[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 7.
ppi16_in_8[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 8.
ppi16_in_9[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 9.
ppi17_in_0[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 0.
ppi17_in_1[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 1.
ppi17_in_10[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 10.
ppi17_in_11[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 11.
ppi17_in_12[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 12.
ppi17_in_13[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 13.
ppi17_in_14[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 14.
ppi17_in_15[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 15.
ppi17_in_16[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 16.
ppi17_in_17[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 17.

Name	Protocol	Type	Description
ppi17_in_18[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 18.
ppi17_in_19[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 19.
ppi17_in_2[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 2.
ppi17_in_20[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 20.
ppi17_in_21[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 21.
ppi17_in_22[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 22.
ppi17_in_23[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 23.
ppi17_in_24[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 24.
ppi17_in_25[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 25.
ppi17_in_26[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 26.
ppi17_in_27[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 27.
ppi17_in_28[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 28.
ppi17_in_29[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 29.
ppi17_in_3[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 3.
ppi17_in_30[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 30.
ppi17_in_31[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 31.
ppi17_in_4[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 4.
ppi17_in_5[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 5.
ppi17_in_6[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 6.
ppi17_in_7[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 7.
ppi17_in_8[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 8.
ppi17_in_9[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 9.
ppi18_in_0[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 0.
ppi18_in_1[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 1.
ppi18_in_10[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 10.
ppi18_in_11[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 11.
ppi18_in_12[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 12.
ppi18_in_13[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 13.
ppi18_in_14[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 14.
ppi18_in_15[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 15.
ppi18_in_16[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 16.
ppi18_in_17[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 17.
ppi18_in_18[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 18.
ppi18_in_19[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 19.
ppi18_in_2[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 2.
ppi18_in_20[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 20.
ppi18_in_21[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 21.
ppi18_in_22[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 22.
ppi18_in_23[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 23.

Name	Protocol	Type	Description
ppi18_in_24[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 24.
ppi18_in_25[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 25.
ppi18_in_26[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 26.
ppi18_in_27[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 27.
ppi18_in_28[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 28.
ppi18_in_29[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 29.
ppi18_in_3[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 3.
ppi18_in_30[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 30.
ppi18_in_31[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 31.
ppi18_in_4[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 4.
ppi18_in_5[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 5.
ppi18_in_6[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 6.
ppi18_in_7[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 7.
ppi18_in_8[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 8.
ppi18_in_9[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 9.
ppi19_in_0[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 0.
ppi19_in_1[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 1.
ppi19_in_10[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 10.
ppi19_in_11[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 11.
ppi19_in_12[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 12.
ppi19_in_13[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 13.
ppi19_in_14[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 14.
ppi19_in_15[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 15.
ppi19_in_16[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 16.
ppi19_in_17[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 17.
ppi19_in_18[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 18.
ppi19_in_19[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 19.
ppi19_in_2[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 2.
ppi19_in_20[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 20.
ppi19_in_21[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 21.
ppi19_in_22[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 22.
ppi19_in_23[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 23.
ppi19_in_24[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 24.
ppi19_in_25[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 25.
ppi19_in_26[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 26.
ppi19_in_27[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 27.
ppi19_in_28[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 28.
ppi19_in_29[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 29.
ppi19_in_3[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 3.

Name	Protocol	Type	Description
ppi19_in_30[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 30.
ppi19_in_31[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 31.
ppi19_in_4[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 4.
ppi19_in_5[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 5.
ppi19_in_6[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 6.
ppi19_in_7[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 7.
ppi19_in_8[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 8.
ppi19_in_9[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 9.
ppi20_in_0[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 0.
ppi20_in_1[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 1.
ppi20_in_10[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 10.
ppi20_in_11[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 11.
ppi20_in_12[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 12.
ppi20_in_13[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 13.
ppi20_in_14[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 14.
ppi20_in_15[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 15.
ppi20_in_16[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 16.
ppi20_in_17[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 17.
ppi20_in_18[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 18.
ppi20_in_19[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 19.
ppi20_in_2[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 2.
ppi20_in_20[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 20.
ppi20_in_21[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 21.
ppi20_in_22[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 22.
ppi20_in_23[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 23.
ppi20_in_24[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 24.
ppi20_in_25[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 25.
ppi20_in_26[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 26.
ppi20_in_27[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 27.
ppi20_in_28[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 28.
ppi20_in_29[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 29.
ppi20_in_3[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 3.
ppi20_in_30[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 30.
ppi20_in_31[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 31.
ppi20_in_4[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 4.
ppi20_in_5[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 5.
ppi20_in_6[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 6.
ppi20_in_7[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 7.
ppi20_in_8[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 8.

Name	Protocol	Type	Description
ppi20_in_9[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 9.
ppi21_in_0[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 0.
ppi21_in_1[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 1.
ppi21_in_10[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 10.
ppi21_in_11[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 11.
ppi21_in_12[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 12.
ppi21_in_13[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 13.
ppi21_in_14[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 14.
ppi21_in_15[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 15.
ppi21_in_16[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 16.
ppi21_in_17[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 17.
ppi21_in_18[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 18.
ppi21_in_19[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 19.
ppi21_in_2[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 2.
ppi21_in_20[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 20.
ppi21_in_21[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 21.
ppi21_in_22[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 22.
ppi21_in_23[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 23.
ppi21_in_24[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 24.
ppi21_in_25[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 25.
ppi21_in_26[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 26.
ppi21_in_27[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 27.
ppi21_in_28[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 28.
ppi21_in_29[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 29.
ppi21_in_3[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 3.
ppi21_in_30[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 30.
ppi21_in_31[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 31.
ppi21_in_4[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 4.
ppi21_in_5[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 5.
ppi21_in_6[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 6.
ppi21_in_7[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 7.
ppi21_in_8[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 8.
ppi21_in_9[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 9.
ppi22_in_0[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 0.
ppi22_in_1[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 1.
ppi22_in_10[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 10.

Name	Protocol	Type	Description
ppi22_in_11[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 11.
ppi22_in_12[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 12.
ppi22_in_13[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 13.
ppi22_in_14[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 14.
ppi22_in_15[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 15.
ppi22_in_16[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 16.
ppi22_in_17[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 17.
ppi22_in_18[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 18.
ppi22_in_19[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 19.
ppi22_in_2[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 2.
ppi22_in_20[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 20.
ppi22_in_21[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 21.
ppi22_in_22[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 22.
ppi22_in_23[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 23.
ppi22_in_24[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 24.
ppi22_in_25[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 25.
ppi22_in_26[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 26.
ppi22_in_27[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 27.
ppi22_in_28[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 28.
ppi22_in_29[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 29.
ppi22_in_3[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 3.
ppi22_in_30[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 30.
ppi22_in_31[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 31.

Name	Protocol	Type	Description
ppi22_in_4[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 4.
ppi22_in_5[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 5.
ppi22_in_6[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 6.
ppi22_in_7[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 7.
ppi22_in_8[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 8.
ppi22_in_9[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 9.
ppi23_in_0[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 0.
ppi23_in_1[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 1.
ppi23_in_10[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 10.
ppi23_in_11[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 11.
ppi23_in_12[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 12.
ppi23_in_13[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 13.
ppi23_in_14[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 14.
ppi23_in_15[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 15.
ppi23_in_16[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 16.
ppi23_in_17[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 17.
ppi23_in_18[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 18.
ppi23_in_19[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 19.
ppi23_in_2[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 2.
ppi23_in_20[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 20.
ppi23_in_21[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 21.
ppi23_in_22[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 22.
ppi23_in_23[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 23.

Name	Protocol	Type	Description
ppi23_in_24[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 24.
ppi23_in_25[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 25.
ppi23_in_26[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 26.
ppi23_in_27[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 27.
ppi23_in_28[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 28.
ppi23_in_29[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 29.
ppi23_in_3[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 3.
ppi23_in_30[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 30.
ppi23_in_31[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 31.
ppi23_in_4[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 4.
ppi23_in_5[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 5.
ppi23_in_6[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 6.
ppi23_in_7[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 7.
ppi23_in_8[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 8.
ppi23_in_9[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 9.
ppi24_in_0[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 0.
ppi24_in_1[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 1.
ppi24_in_10[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 10.
ppi24_in_11[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 11.
ppi24_in_12[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 12.
ppi24_in_13[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 13.
ppi24_in_14[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 14.
ppi24_in_15[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 15.

Name	Protocol	Type	Description
ppi24_in_16[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 16.
ppi24_in_17[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 17.
ppi24_in_18[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 18.
ppi24_in_19[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 19.
ppi24_in_2[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 2.
ppi24_in_20[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 20.
ppi24_in_21[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 21.
ppi24_in_22[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 22.
ppi24_in_23[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 23.
ppi24_in_24[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 24.
ppi24_in_25[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 25.
ppi24_in_26[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 26.
ppi24_in_27[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 27.
ppi24_in_28[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 28.
ppi24_in_29[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 29.
ppi24_in_3[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 3.
ppi24_in_30[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 30.
ppi24_in_31[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 31.
ppi24_in_4[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 4.
ppi24_in_5[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 5.
ppi24_in_6[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 6.
ppi24_in_7[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 7.
ppi24_in_8[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 8.

Name	Protocol	Type	Description
ppi24_in_9[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 9.
ppi25_in_0[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 0.
ppi25_in_1[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 1.
ppi25_in_10[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 10.
ppi25_in_11[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 11.
ppi25_in_12[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 12.
ppi25_in_13[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 13.
ppi25_in_14[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 14.
ppi25_in_15[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 15.
ppi25_in_16[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 16.
ppi25_in_17[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 17.
ppi25_in_18[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 18.
ppi25_in_19[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 19.
ppi25_in_2[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 2.
ppi25_in_20[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 20.
ppi25_in_21[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 21.
ppi25_in_22[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 22.
ppi25_in_23[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 23.
ppi25_in_24[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 24.
ppi25_in_25[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 25.
ppi25_in_26[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 26.
ppi25_in_27[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 27.
ppi25_in_28[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 28.

Name	Protocol	Type	Description
ppi25_in_29[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 29.
ppi25_in_3[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 3.
ppi25_in_30[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 30.
ppi25_in_31[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 31.
ppi25_in_4[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 4.
ppi25_in_5[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 5.
ppi25_in_6[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 6.
ppi25_in_7[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 7.
ppi25_in_8[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 8.
ppi25_in_9[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 9.
ppi26_in_0[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 0.
ppi26_in_1[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 1.
ppi26_in_10[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 10.
ppi26_in_11[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 11.
ppi26_in_12[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 12.
ppi26_in_13[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 13.
ppi26_in_14[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 14.
ppi26_in_15[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 15.
ppi26_in_16[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 16.
ppi26_in_17[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 17.
ppi26_in_18[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 18.
ppi26_in_19[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 19.
ppi26_in_2[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 2.
ppi26_in_20[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 20.
ppi26_in_21[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 21.
ppi26_in_22[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 22.
ppi26_in_23[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 23.
ppi26_in_24[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 24.
ppi26_in_25[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 25.
ppi26_in_26[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 26.
ppi26_in_27[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 27.
ppi26_in_28[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 28.

Name	Protocol	Type	Description
ppi26_in_29[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 29.
ppi26_in_3[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 3.
ppi26_in_30[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 30.
ppi26_in_31[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 31.
ppi26_in_4[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 4.
ppi26_in_5[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 5.
ppi26_in_6[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 6.
ppi26_in_7[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 7.
ppi26_in_8[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 8.
ppi26_in_9[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 9.
ppi27_in_0[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 0.
ppi27_in_1[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 1.
ppi27_in_10[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 10.
ppi27_in_11[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 11.
ppi27_in_12[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 12.
ppi27_in_13[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 13.
ppi27_in_14[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 14.
ppi27_in_15[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 15.
ppi27_in_16[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 16.
ppi27_in_17[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 17.
ppi27_in_18[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 18.
ppi27_in_19[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 19.
ppi27_in_2[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 2.
ppi27_in_20[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 20.
ppi27_in_21[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 21.
ppi27_in_22[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 22.
ppi27_in_23[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 23.
ppi27_in_24[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 24.
ppi27_in_25[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 25.
ppi27_in_26[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 26.
ppi27_in_27[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 27.
ppi27_in_28[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 28.
ppi27_in_29[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 29.
ppi27_in_3[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 3.
ppi27_in_30[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 30.
ppi27_in_31[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 31.
ppi27_in_4[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 4.
ppi27_in_5[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 5.
ppi27_in_6[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 6.

Name	Protocol	Type	Description
ppi27_in_7[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 7.
ppi27_in_8[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 8.
ppi27_in_9[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 9.
ppi28_in_0[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 0.
ppi28_in_1[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 1.
ppi28_in_10[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 10.
ppi28_in_11[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 11.
ppi28_in_12[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 12.
ppi28_in_13[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 13.
ppi28_in_14[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 14.
ppi28_in_15[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 15.
ppi28_in_16[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 16.
ppi28_in_17[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 17.
ppi28_in_18[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 18.
ppi28_in_19[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 19.
ppi28_in_2[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 2.
ppi28_in_20[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 20.
ppi28_in_21[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 21.
ppi28_in_22[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 22.
ppi28_in_23[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 23.
ppi28_in_24[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 24.
ppi28_in_25[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 25.
ppi28_in_26[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 26.
ppi28_in_27[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 27.
ppi28_in_28[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 28.
ppi28_in_29[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 29.
ppi28_in_3[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 3.
ppi28_in_30[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 30.
ppi28_in_31[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 31.
ppi28_in_4[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 4.
ppi28_in_5[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 5.
ppi28_in_6[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 6.
ppi28_in_7[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 7.
ppi28_in_8[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 8.
ppi28_in_9[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 9.
ppi29_in_0[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 0.
ppi29_in_1[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 1.
ppi29_in_10[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 10.
ppi29_in_11[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 11.

Name	Protocol	Type	Description
ppi29_in_12[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 12.
ppi29_in_13[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 13.
ppi29_in_14[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 14.
ppi29_in_15[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 15.
ppi29_in_16[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 16.
ppi29_in_17[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 17.
ppi29_in_18[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 18.
ppi29_in_19[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 19.
ppi29_in_2[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 2.
ppi29_in_20[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 20.
ppi29_in_21[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 21.
ppi29_in_22[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 22.
ppi29_in_23[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 23.
ppi29_in_24[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 24.
ppi29_in_25[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 25.
ppi29_in_26[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 26.
ppi29_in_27[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 27.
ppi29_in_28[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 28.
ppi29_in_29[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 29.
ppi29_in_3[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 3.
ppi29_in_30[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 30.
ppi29_in_31[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 31.
ppi29_in_4[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 4.
ppi29_in_5[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 5.
ppi29_in_6[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 6.
ppi29_in_7[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 7.
ppi29_in_8[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 8.
ppi29_in_9[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 9.
ppi30_in_0[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 0.
ppi30_in_1[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 1.
ppi30_in_10[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 10.
ppi30_in_11[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 11.
ppi30_in_12[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 12.
ppi30_in_13[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 13.

Name	Protocol	Type	Description
ppi30_in_14[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 14.
ppi30_in_15[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 15.
ppi30_in_16[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 16.
ppi30_in_17[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 17.
ppi30_in_18[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 18.
ppi30_in_19[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 19.
ppi30_in_2[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 2.
ppi30_in_20[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 20.
ppi30_in_21[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 21.
ppi30_in_22[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 22.
ppi30_in_23[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 23.
ppi30_in_24[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 24.
ppi30_in_25[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 25.
ppi30_in_26[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 26.
ppi30_in_27[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 27.
ppi30_in_28[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 28.
ppi30_in_29[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 29.
ppi30_in_3[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 3.
ppi30_in_30[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 30.
ppi30_in_31[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 31.
ppi30_in_4[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 4.
ppi30_in_5[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 5.
ppi30_in_6[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 6.

Name	Protocol	Type	Description
ppi30_in_7[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 7.
ppi30_in_8[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 8.
ppi30_in_9[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 9.
ppi31_in_0[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 0.
ppi31_in_1[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 1.
ppi31_in_10[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 10.
ppi31_in_11[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 11.
ppi31_in_12[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 12.
ppi31_in_13[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 13.
ppi31_in_14[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 14.
ppi31_in_15[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 15.
ppi31_in_16[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 16.
ppi31_in_17[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 17.
ppi31_in_18[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 18.
ppi31_in_19[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 19.
ppi31_in_2[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 2.
ppi31_in_20[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 20.
ppi31_in_21[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 21.
ppi31_in_22[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 22.
ppi31_in_23[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 23.
ppi31_in_24[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 24.
ppi31_in_25[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 25.
ppi31_in_26[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 26.
ppi31_in_27[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 27.
ppi31_in_28[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 28.
ppi31_in_29[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 29.
ppi31_in_3[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 3.
ppi31_in_30[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 30.
ppi31_in_31[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 31.
ppi31_in_4[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 4.
ppi31_in_5[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 5.
ppi31_in_6[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 6.
ppi31_in_7[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 7.
ppi31_in_8[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 8.
ppi31_in_9[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 9.
pvbuses_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbuses_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.

Name	Protocol	Type	Description
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Reset.
spi_in[988]	Signal	Slave	Shared peripheral interrupts.
wake_request_0[8]	Signal	Master	Power management outputs of cluster 0.
wake_request_1[8]	Signal	Master	Power management outputs of cluster 1.
wake_request_10[8]	Signal	Master	Power management outputs of cluster 10.
wake_request_11[8]	Signal	Master	Power management outputs of cluster 11.
wake_request_12[8]	Signal	Master	Power management outputs of cluster 12.
wake_request_13[8]	Signal	Master	Power management outputs of cluster 13.
wake_request_14[8]	Signal	Master	Power management outputs of cluster 14.
wake_request_15[8]	Signal	Master	Power management outputs of cluster 15.
wake_request_16[8]	Signal	Master	Power management outputs of cluster 16.
wake_request_17[8]	Signal	Master	Power management outputs of cluster 17.
wake_request_18[8]	Signal	Master	Power management outputs of cluster 18.
wake_request_19[8]	Signal	Master	Power management outputs of cluster 19.
wake_request_2[8]	Signal	Master	Power management outputs of cluster 2.
wake_request_20[8]	Signal	Master	Power management outputs of cluster 20.
wake_request_21[8]	Signal	Master	Power management outputs of cluster 21.
wake_request_22[8]	Signal	Master	Power management outputs of cluster 22.
wake_request_23[8]	Signal	Master	Power management outputs of cluster 23.
wake_request_24[8]	Signal	Master	Power management outputs of cluster 24.
wake_request_25[8]	Signal	Master	Power management outputs of cluster 25.
wake_request_26[8]	Signal	Master	Power management outputs of cluster 26.
wake_request_27[8]	Signal	Master	Power management outputs of cluster 27.
wake_request_28[8]	Signal	Master	Power management outputs of cluster 28.
wake_request_29[8]	Signal	Master	Power management outputs of cluster 29.
wake_request_3[8]	Signal	Master	Power management outputs of cluster 3.
wake_request_30[8]	Signal	Master	Power management outputs of cluster 30.
wake_request_31[8]	Signal	Master	Power management outputs of cluster 31.
wake_request_4[8]	Signal	Master	Power management outputs of cluster 4.
wake_request_5[8]	Signal	Master	Power management outputs of cluster 5.
wake_request_6[8]	Signal	Master	Power management outputs of cluster 6.
wake_request_7[8]	Signal	Master	Power management outputs of cluster 7.
wake_request_8[8]	Signal	Master	Power management outputs of cluster 8.
wake_request_9[8]	Signal	Master	Power management outputs of cluster 9.

Parameters for GIC500

GICD_ITARGETSR-RAZWI

If true, the GICD_ITARGETS registers are RAZ/WI

Type

bool

Default value

0x0

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

ITS-threaded-command-queue

Enable execution of ITS commands in a separate thread which is sometimes required for cosimulation

Type

bool

Default value

0x1

SPI-count

Number of SPIs that are implemented.

Type

int

Default value

0xe0

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

cpus_per_cluster_0

Number of cores within cluster 0

Type

int

Default value

0x1

cpus_per_cluster_1

Number of cores within cluster 1

Type

int

Default value

0x1

cpus_per_cluster_10

Number of cores within cluster 10

Type

int

Default value

0x1

cpus_per_cluster_11

Number of cores within cluster 11

Type

int

Default value

0x1

cpus_per_cluster_12

Number of cores within cluster 12

Type

int

Default value

0x1

cpus_per_cluster_13

Number of cores within cluster 13

Type

int

Default value

0x1

cpus_per_cluster_14

Number of cores within cluster 14

Type

int

Default value

0x1

cpus_per_cluster_15

Number of cores within cluster 15

Type

int

Default value

0x1

cpus_per_cluster_16

Number of cores within cluster 16

Type

int

Default value

0x1

cpus_per_cluster_17

Number of cores within cluster 17

Type

int

Default value

0x1

cpus_per_cluster_18

Number of cores within cluster 18

Type

int

Default value

0x1

cpus_per_cluster_19

Number of cores within cluster 19

Type

int

Default value

0x1

cpus_per_cluster_2

Number of cores within cluster 2

Type

int

Default value

0x1

cpus_per_cluster_20

Number of cores within cluster 20

Type

int

Default value

0x1

cpus_per_cluster_21

Number of cores within cluster 21

Type

int

Default value

0x1

cpus_per_cluster_22

Number of cores within cluster 22

Type

int

Default value

0x1

cpus_per_cluster_23

Number of cores within cluster 23

Type

int

Default value

0x1

cpus_per_cluster_24

Number of cores within cluster 24

Type

int

Default value

0x1

cpus_per_cluster_25

Number of cores within cluster 25

Type

int

Default value

0x1

cpus_per_cluster_26

Number of cores within cluster 26

Type

int

Default value

0x1

cpus_per_cluster_27

Number of cores within cluster 27

Type

int

Default value

0x1

cpus_per_cluster_28

Number of cores within cluster 28

Type

int

Default value

0x1

cpus_per_cluster_29

Number of cores within cluster 29

Type

int

Default value

0x1

cpus_per_cluster_3

Number of cores within cluster 3

Type

int

Default value

0x1

cpus_per_cluster_30

Number of cores within cluster 30

Type

int

Default value

0x1

cpus_per_cluster_31

Number of cores within cluster 31

Type

int

Default value

0x1

cpus_per_cluster_4

Number of cores within cluster 4

Type

int

Default value

0x1

cpus_per_cluster_5

Number of cores within cluster 5

Type

int

Default value

0x1

cpus_per_cluster_6

Number of cores within cluster 6

Type

int

Default value

0x1

cpus_per_cluster_7

Number of cores within cluster 7

Type

int

Default value

0x1

cpus_per_cluster_8

Number of cores within cluster 8

Type

int

Default value

0x1

cpus_per_cluster_9

Number of cores within cluster 9

Type

int

Default value

0x1

delay-ITS-accesses

Delay accesses from the ITS until GICR_SYNCRR is read.

Type

bool

Default value

0x1

delay-redistributor-accesses

Delay memory accesses from the redistributor until GICR_SYNCRR is read.

Type

bool

Default value

0x1

enable_protocol_checking

Enable/disable protocol checking at cpu interface

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

has-two-security-states

If true, has two security states

Type

bool

Default value

0x1

num_clusters

Number of implemented affinity level1 clusters

Type

int

Default value

0x1

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

redistributor-threaded-sync

Enable execution of redistributor delayed transactions in a separate thread which is sometimes required for cosimulation

Type

bool

Default value

0x1

reg-base

GIC500 base address

Type

int

Default value

0x10000000

using-generated-memorymap

Use the generated memorymap for the GIC500 and warn if superfluous parameters are passed

Type

bool

Default value

0x1

wakeup-on-reset

Go against specification and start redistributors in woken-up state at reset. This allows software that was written for previous versions of the GICv3 specification to work correctly. This should not be used for production code or when the distributor is used separately from the core fast model.

Type

bool

Default value

0x0

3.10.29 GIC500_ClusterPorts

GIC500 Component for distribution of interrupts. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-938: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC500_ClusterPorts

This model has the following Iris instances:

Table 3-939: GIC500_ClusterPorts Iris instances

InstanceName	ComponentName
GIC500_ClusterPorts	GIC_IRI
GIC500_ClusterPorts.ITS0	GICv3InterruptTranslationService
GIC500_ClusterPorts.rd_0	GICv3RedistributorInternal
GIC500_ClusterPorts.rd_0_0	GICv3RedistributorInternal
GIC500_ClusterPorts.rd_0_0_0	GICv3RedistributorInternal
GIC500_ClusterPorts.rd_0_0_0_0	GICv3Redistributor
GIC500_ClusterPorts.rd_tl	GICv3Distributor

This model has the following MTI trace components:

Table 3-940: GIC500_ClusterPorts MTI instances

InstanceName	ComponentName
GIC500_ClusterPorts	GICv3IRI

InstanceName	ComponentName
GIC500_ClusterPorts.ITS0	GICv3InterruptTranslationService
GIC500_ClusterPorts.rd_0	GICv3RedistributorInternal
GIC500_ClusterPorts.rd_0_0	GICv3RedistributorInternal
GIC500_ClusterPorts.rd_0_0_0	GICv3RedistributorInternal
GIC500_ClusterPorts.rd_0_0_0_0	GICv3Redistributor
GIC500_ClusterPorts.rd_t1	GICv3Distributor

GIC500_ClusterPorts contains the following CADI targets:

- GIC500

Ports for GIC500_ClusterPorts

Table 3-941: Ports

Name	Protocol	Type	Description
cfgsdisable	Signal	Slave	Disable some SPIs signal.
cpu_active_0[8]	Signal	Slave	cpu_active pins of cluster 0.
cpu_active_1[8]	Signal	Slave	cpu_active pins of cluster 1.
cpu_active_10[8]	Signal	Slave	cpu_active pins of cluster 10.
cpu_active_11[8]	Signal	Slave	cpu_active pins of cluster 11.
cpu_active_12[8]	Signal	Slave	cpu_active pins of cluster 12.
cpu_active_13[8]	Signal	Slave	cpu_active pins of cluster 13.
cpu_active_14[8]	Signal	Slave	cpu_active pins of cluster 14.
cpu_active_15[8]	Signal	Slave	cpu_active pins of cluster 15.
cpu_active_16[8]	Signal	Slave	cpu_active pins of cluster 16.
cpu_active_17[8]	Signal	Slave	cpu_active pins of cluster 17.
cpu_active_18[8]	Signal	Slave	cpu_active pins of cluster 18.
cpu_active_19[8]	Signal	Slave	cpu_active pins of cluster 19.
cpu_active_2[8]	Signal	Slave	cpu_active pins of cluster 2.
cpu_active_20[8]	Signal	Slave	cpu_active pins of cluster 20.
cpu_active_21[8]	Signal	Slave	cpu_active pins of cluster 21.
cpu_active_22[8]	Signal	Slave	cpu_active pins of cluster 22.
cpu_active_23[8]	Signal	Slave	cpu_active pins of cluster 23.
cpu_active_24[8]	Signal	Slave	cpu_active pins of cluster 24.
cpu_active_25[8]	Signal	Slave	cpu_active pins of cluster 25.
cpu_active_26[8]	Signal	Slave	cpu_active pins of cluster 26.
cpu_active_27[8]	Signal	Slave	cpu_active pins of cluster 27.
cpu_active_28[8]	Signal	Slave	cpu_active pins of cluster 28.
cpu_active_29[8]	Signal	Slave	cpu_active pins of cluster 29.
cpu_active_3[8]	Signal	Slave	cpu_active pins of cluster 3.
cpu_active_30[8]	Signal	Slave	cpu_active pins of cluster 30.

Name	Protocol	Type	Description
cpu_active_31[8]	Signal	Slave	cpu_active pins of cluster 31.
cpu_active_4[8]	Signal	Slave	cpu_active pins of cluster 4.
cpu_active_5[8]	Signal	Slave	cpu_active pins of cluster 5.
cpu_active_6[8]	Signal	Slave	cpu_active pins of cluster 6.
cpu_active_7[8]	Signal	Slave	cpu_active pins of cluster 7.
cpu_active_8[8]	Signal	Slave	cpu_active pins of cluster 8.
cpu_active_9[8]	Signal	Slave	cpu_active pins of cluster 9.
po_reset	Signal	Slave	Power on reset.
ppi16_in_0[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 0.
ppi16_in_1[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 1.
ppi16_in_10[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 10.
ppi16_in_11[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 11.
ppi16_in_12[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 12.
ppi16_in_13[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 13.
ppi16_in_14[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 14.
ppi16_in_15[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 15.
ppi16_in_16[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 16.
ppi16_in_17[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 17.
ppi16_in_18[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 18.
ppi16_in_19[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 19.
ppi16_in_2[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 2.
ppi16_in_20[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 20.
ppi16_in_21[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 21.
ppi16_in_22[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 22.
ppi16_in_23[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 23.
ppi16_in_24[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 24.
ppi16_in_25[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 25.
ppi16_in_26[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 26.
ppi16_in_27[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 27.
ppi16_in_28[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 28.
ppi16_in_29[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 29.
ppi16_in_3[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 3.
ppi16_in_30[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 30.
ppi16_in_31[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 31.
ppi16_in_4[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 4.
ppi16_in_5[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 5.
ppi16_in_6[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 6.
ppi16_in_7[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 7.
ppi16_in_8[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 8.

Name	Protocol	Type	Description
ppi16_in_9[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 9.
ppi17_in_0[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 0.
ppi17_in_1[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 1.
ppi17_in_10[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 10.
ppi17_in_11[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 11.
ppi17_in_12[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 12.
ppi17_in_13[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 13.
ppi17_in_14[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 14.
ppi17_in_15[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 15.
ppi17_in_16[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 16.
ppi17_in_17[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 17.
ppi17_in_18[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 18.
ppi17_in_19[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 19.
ppi17_in_2[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 2.
ppi17_in_20[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 20.
ppi17_in_21[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 21.
ppi17_in_22[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 22.
ppi17_in_23[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 23.
ppi17_in_24[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 24.
ppi17_in_25[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 25.
ppi17_in_26[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 26.
ppi17_in_27[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 27.
ppi17_in_28[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 28.
ppi17_in_29[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 29.
ppi17_in_3[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 3.
ppi17_in_30[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 30.
ppi17_in_31[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 31.
ppi17_in_4[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 4.
ppi17_in_5[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 5.
ppi17_in_6[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 6.
ppi17_in_7[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 7.
ppi17_in_8[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 8.
ppi17_in_9[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 9.
ppi18_in_0[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 0.
ppi18_in_1[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 1.
ppi18_in_10[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 10.
ppi18_in_11[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 11.
ppi18_in_12[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 12.
ppi18_in_13[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 13.

Name	Protocol	Type	Description
ppi18_in_14[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 14.
ppi18_in_15[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 15.
ppi18_in_16[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 16.
ppi18_in_17[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 17.
ppi18_in_18[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 18.
ppi18_in_19[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 19.
ppi18_in_2[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 2.
ppi18_in_20[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 20.
ppi18_in_21[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 21.
ppi18_in_22[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 22.
ppi18_in_23[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 23.
ppi18_in_24[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 24.
ppi18_in_25[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 25.
ppi18_in_26[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 26.
ppi18_in_27[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 27.
ppi18_in_28[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 28.
ppi18_in_29[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 29.
ppi18_in_3[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 3.
ppi18_in_30[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 30.
ppi18_in_31[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 31.
ppi18_in_4[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 4.
ppi18_in_5[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 5.
ppi18_in_6[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 6.
ppi18_in_7[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 7.
ppi18_in_8[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 8.
ppi18_in_9[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 9.
ppi19_in_0[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 0.
ppi19_in_1[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 1.
ppi19_in_10[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 10.
ppi19_in_11[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 11.
ppi19_in_12[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 12.
ppi19_in_13[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 13.
ppi19_in_14[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 14.
ppi19_in_15[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 15.
ppi19_in_16[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 16.
ppi19_in_17[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 17.
ppi19_in_18[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 18.
ppi19_in_19[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 19.
ppi19_in_2[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 2.

Name	Protocol	Type	Description
ppi19_in_20[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 20.
ppi19_in_21[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 21.
ppi19_in_22[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 22.
ppi19_in_23[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 23.
ppi19_in_24[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 24.
ppi19_in_25[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 25.
ppi19_in_26[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 26.
ppi19_in_27[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 27.
ppi19_in_28[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 28.
ppi19_in_29[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 29.
ppi19_in_3[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 3.
ppi19_in_30[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 30.
ppi19_in_31[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 31.
ppi19_in_4[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 4.
ppi19_in_5[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 5.
ppi19_in_6[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 6.
ppi19_in_7[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 7.
ppi19_in_8[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 8.
ppi19_in_9[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 9.
ppi20_in_0[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 0.
ppi20_in_1[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 1.
ppi20_in_10[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 10.
ppi20_in_11[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 11.
ppi20_in_12[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 12.
ppi20_in_13[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 13.
ppi20_in_14[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 14.
ppi20_in_15[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 15.
ppi20_in_16[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 16.
ppi20_in_17[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 17.
ppi20_in_18[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 18.
ppi20_in_19[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 19.
ppi20_in_2[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 2.
ppi20_in_20[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 20.
ppi20_in_21[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 21.
ppi20_in_22[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 22.
ppi20_in_23[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 23.
ppi20_in_24[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 24.
ppi20_in_25[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 25.
ppi20_in_26[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 26.

Name	Protocol	Type	Description
ppi20_in_27[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 27.
ppi20_in_28[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 28.
ppi20_in_29[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 29.
ppi20_in_3[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 3.
ppi20_in_30[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 30.
ppi20_in_31[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 31.
ppi20_in_4[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 4.
ppi20_in_5[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 5.
ppi20_in_6[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 6.
ppi20_in_7[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 7.
ppi20_in_8[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 8.
ppi20_in_9[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 9.
ppi21_in_0[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 0.
ppi21_in_1[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 1.
ppi21_in_10[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 10.
ppi21_in_11[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 11.
ppi21_in_12[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 12.
ppi21_in_13[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 13.
ppi21_in_14[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 14.
ppi21_in_15[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 15.
ppi21_in_16[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 16.
ppi21_in_17[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 17.
ppi21_in_18[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 18.
ppi21_in_19[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 19.
ppi21_in_2[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 2.
ppi21_in_20[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 20.
ppi21_in_21[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 21.
ppi21_in_22[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 22.
ppi21_in_23[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 23.
ppi21_in_24[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 24.
ppi21_in_25[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 25.
ppi21_in_26[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 26.
ppi21_in_27[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 27.
ppi21_in_28[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 28.
ppi21_in_29[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 29.
ppi21_in_3[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 3.
ppi21_in_30[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 30.
ppi21_in_31[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 31.
ppi21_in_4[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 4.

Name	Protocol	Type	Description
ppi21_in_5[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 5.
ppi21_in_6[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 6.
ppi21_in_7[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 7.
ppi21_in_8[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 8.
ppi21_in_9[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 9.
ppi22_in_0[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 0.
ppi22_in_1[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 1.
ppi22_in_10[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 10.
ppi22_in_11[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 11.
ppi22_in_12[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 12.
ppi22_in_13[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 13.
ppi22_in_14[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 14.
ppi22_in_15[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 15.
ppi22_in_16[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 16.
ppi22_in_17[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 17.
ppi22_in_18[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 18.
ppi22_in_19[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 19.
ppi22_in_2[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 2.
ppi22_in_20[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 20.
ppi22_in_21[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 21.
ppi22_in_22[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 22.
ppi22_in_23[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 23.
ppi22_in_24[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 24.
ppi22_in_25[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 25.
ppi22_in_26[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 26.

Name	Protocol	Type	Description
ppi22_in_27[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 27.
ppi22_in_28[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 28.
ppi22_in_29[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 29.
ppi22_in_3[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 3.
ppi22_in_30[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 30.
ppi22_in_31[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 31.
ppi22_in_4[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 4.
ppi22_in_5[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 5.
ppi22_in_6[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 6.
ppi22_in_7[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 7.
ppi22_in_8[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 8.
ppi22_in_9[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 9.
ppi23_in_0[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 0.
ppi23_in_1[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 1.
ppi23_in_10[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 10.
ppi23_in_11[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 11.
ppi23_in_12[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 12.
ppi23_in_13[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 13.
ppi23_in_14[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 14.
ppi23_in_15[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 15.
ppi23_in_16[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 16.
ppi23_in_17[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 17.
ppi23_in_18[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 18.

Name	Protocol	Type	Description
ppi23_in_19[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 19.
ppi23_in_2[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 2.
ppi23_in_20[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 20.
ppi23_in_21[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 21.
ppi23_in_22[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 22.
ppi23_in_23[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 23.
ppi23_in_24[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 24.
ppi23_in_25[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 25.
ppi23_in_26[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 26.
ppi23_in_27[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 27.
ppi23_in_28[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 28.
ppi23_in_29[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 29.
ppi23_in_3[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 3.
ppi23_in_30[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 30.
ppi23_in_31[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 31.
ppi23_in_4[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 4.
ppi23_in_5[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 5.
ppi23_in_6[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 6.
ppi23_in_7[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 7.
ppi23_in_8[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 8.
ppi23_in_9[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 9.
ppi24_in_0[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 0.
ppi24_in_1[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 1.

Name	Protocol	Type	Description
ppi24_in_10[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 10.
ppi24_in_11[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 11.
ppi24_in_12[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 12.
ppi24_in_13[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 13.
ppi24_in_14[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 14.
ppi24_in_15[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 15.
ppi24_in_16[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 16.
ppi24_in_17[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 17.
ppi24_in_18[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 18.
ppi24_in_19[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 19.
ppi24_in_2[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 2.
ppi24_in_20[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 20.
ppi24_in_21[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 21.
ppi24_in_22[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 22.
ppi24_in_23[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 23.
ppi24_in_24[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 24.
ppi24_in_25[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 25.
ppi24_in_26[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 26.
ppi24_in_27[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 27.
ppi24_in_28[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 28.
ppi24_in_29[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 29.
ppi24_in_3[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 3.
ppi24_in_30[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 30.

Name	Protocol	Type	Description
ppi24_in_31[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 31.
ppi24_in_4[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 4.
ppi24_in_5[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 5.
ppi24_in_6[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 6.
ppi24_in_7[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 7.
ppi24_in_8[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 8.
ppi24_in_9[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 9.
ppi25_in_0[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 0.
ppi25_in_1[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 1.
ppi25_in_10[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 10.
ppi25_in_11[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 11.
ppi25_in_12[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 12.
ppi25_in_13[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 13.
ppi25_in_14[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 14.
ppi25_in_15[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 15.
ppi25_in_16[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 16.
ppi25_in_17[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 17.
ppi25_in_18[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 18.
ppi25_in_19[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 19.
ppi25_in_2[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 2.
ppi25_in_20[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 20.
ppi25_in_21[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 21.
ppi25_in_22[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 22.

Name	Protocol	Type	Description
ppi25_in_23[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 23.
ppi25_in_24[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 24.
ppi25_in_25[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 25.
ppi25_in_26[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 26.
ppi25_in_27[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 27.
ppi25_in_28[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 28.
ppi25_in_29[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 29.
ppi25_in_3[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 3.
ppi25_in_30[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 30.
ppi25_in_31[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 31.
ppi25_in_4[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 4.
ppi25_in_5[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 5.
ppi25_in_6[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 6.
ppi25_in_7[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 7.
ppi25_in_8[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 8.
ppi25_in_9[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 9.
ppi26_in_0[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 0.
ppi26_in_1[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 1.
ppi26_in_10[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 10.
ppi26_in_11[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 11.
ppi26_in_12[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 12.
ppi26_in_13[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 13.
ppi26_in_14[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 14.
ppi26_in_15[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 15.
ppi26_in_16[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 16.
ppi26_in_17[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 17.
ppi26_in_18[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 18.
ppi26_in_19[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 19.

Name	Protocol	Type	Description
ppi26_in_2[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 2.
ppi26_in_20[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 20.
ppi26_in_21[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 21.
ppi26_in_22[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 22.
ppi26_in_23[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 23.
ppi26_in_24[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 24.
ppi26_in_25[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 25.
ppi26_in_26[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 26.
ppi26_in_27[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 27.
ppi26_in_28[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 28.
ppi26_in_29[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 29.
ppi26_in_3[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 3.
ppi26_in_30[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 30.
ppi26_in_31[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 31.
ppi26_in_4[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 4.
ppi26_in_5[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 5.
ppi26_in_6[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 6.
ppi26_in_7[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 7.
ppi26_in_8[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 8.
ppi26_in_9[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 9.
ppi27_in_0[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 0.
ppi27_in_1[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 1.
ppi27_in_10[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 10.
ppi27_in_11[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 11.
ppi27_in_12[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 12.
ppi27_in_13[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 13.
ppi27_in_14[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 14.
ppi27_in_15[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 15.
ppi27_in_16[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 16.
ppi27_in_17[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 17.
ppi27_in_18[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 18.
ppi27_in_19[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 19.
ppi27_in_2[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 2.
ppi27_in_20[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 20.
ppi27_in_21[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 21.
ppi27_in_22[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 22.
ppi27_in_23[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 23.
ppi27_in_24[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 24.
ppi27_in_25[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 25.

Name	Protocol	Type	Description
ppi27_in_26[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 26.
ppi27_in_27[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 27.
ppi27_in_28[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 28.
ppi27_in_29[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 29.
ppi27_in_3[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 3.
ppi27_in_30[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 30.
ppi27_in_31[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 31.
ppi27_in_4[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 4.
ppi27_in_5[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 5.
ppi27_in_6[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 6.
ppi27_in_7[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 7.
ppi27_in_8[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 8.
ppi27_in_9[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 9.
ppi28_in_0[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 0.
ppi28_in_1[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 1.
ppi28_in_10[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 10.
ppi28_in_11[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 11.
ppi28_in_12[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 12.
ppi28_in_13[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 13.
ppi28_in_14[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 14.
ppi28_in_15[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 15.
ppi28_in_16[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 16.
ppi28_in_17[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 17.
ppi28_in_18[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 18.
ppi28_in_19[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 19.
ppi28_in_2[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 2.
ppi28_in_20[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 20.
ppi28_in_21[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 21.
ppi28_in_22[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 22.
ppi28_in_23[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 23.
ppi28_in_24[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 24.
ppi28_in_25[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 25.
ppi28_in_26[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 26.
ppi28_in_27[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 27.
ppi28_in_28[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 28.
ppi28_in_29[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 29.
ppi28_in_3[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 3.
ppi28_in_30[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 30.
ppi28_in_31[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 31.

Name	Protocol	Type	Description
ppi28_in_4[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 4.
ppi28_in_5[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 5.
ppi28_in_6[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 6.
ppi28_in_7[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 7.
ppi28_in_8[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 8.
ppi28_in_9[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 9.
ppi29_in_0[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 0.
ppi29_in_1[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 1.
ppi29_in_10[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 10.
ppi29_in_11[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 11.
ppi29_in_12[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 12.
ppi29_in_13[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 13.
ppi29_in_14[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 14.
ppi29_in_15[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 15.
ppi29_in_16[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 16.
ppi29_in_17[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 17.
ppi29_in_18[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 18.
ppi29_in_19[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 19.
ppi29_in_2[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 2.
ppi29_in_20[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 20.
ppi29_in_21[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 21.
ppi29_in_22[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 22.
ppi29_in_23[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 23.
ppi29_in_24[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 24.
ppi29_in_25[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 25.
ppi29_in_26[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 26.
ppi29_in_27[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 27.
ppi29_in_28[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 28.
ppi29_in_29[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 29.
ppi29_in_3[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 3.
ppi29_in_30[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 30.
ppi29_in_31[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 31.
ppi29_in_4[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 4.
ppi29_in_5[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 5.
ppi29_in_6[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 6.
ppi29_in_7[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 7.
ppi29_in_8[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 8.
ppi29_in_9[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 9.

Name	Protocol	Type	Description
ppi30_in_0[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 0.
ppi30_in_1[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 1.
ppi30_in_10[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 10.
ppi30_in_11[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 11.
ppi30_in_12[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 12.
ppi30_in_13[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 13.
ppi30_in_14[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 14.
ppi30_in_15[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 15.
ppi30_in_16[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 16.
ppi30_in_17[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 17.
ppi30_in_18[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 18.
ppi30_in_19[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 19.
ppi30_in_2[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 2.
ppi30_in_20[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 20.
ppi30_in_21[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 21.
ppi30_in_22[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 22.
ppi30_in_23[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 23.
ppi30_in_24[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 24.
ppi30_in_25[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 25.
ppi30_in_26[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 26.
ppi30_in_27[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 27.
ppi30_in_28[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 28.
ppi30_in_29[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 29.

Name	Protocol	Type	Description
ppi30_in_3[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 3.
ppi30_in_30[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 30.
ppi30_in_31[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 31.
ppi30_in_4[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 4.
ppi30_in_5[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 5.
ppi30_in_6[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 6.
ppi30_in_7[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 7.
ppi30_in_8[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 8.
ppi30_in_9[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 9.
ppi31_in_0[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 0.
ppi31_in_1[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 1.
ppi31_in_10[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 10.
ppi31_in_11[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 11.
ppi31_in_12[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 12.
ppi31_in_13[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 13.
ppi31_in_14[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 14.
ppi31_in_15[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 15.
ppi31_in_16[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 16.
ppi31_in_17[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 17.
ppi31_in_18[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 18.
ppi31_in_19[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 19.
ppi31_in_2[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 2.
ppi31_in_20[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 20.
ppi31_in_21[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 21.
ppi31_in_22[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 22.
ppi31_in_23[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 23.
ppi31_in_24[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 24.
ppi31_in_25[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 25.
ppi31_in_26[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 26.
ppi31_in_27[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 27.
ppi31_in_28[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 28.
ppi31_in_29[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 29.
ppi31_in_3[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 3.

Name	Protocol	Type	Description
ppi31_in_30[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 30.
ppi31_in_31[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 31.
ppi31_in_4[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 4.
ppi31_in_5[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 5.
ppi31_in_6[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 6.
ppi31_in_7[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 7.
ppi31_in_8[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 8.
ppi31_in_9[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 9.
pvbus_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_0[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 0
redistributor_1[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 1.
redistributor_10[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 10.
redistributor_11[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 11.
redistributor_12[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 12.
redistributor_13[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 13.
redistributor_14[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 14.
redistributor_15[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 15.
redistributor_16[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 16.
redistributor_17[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 17.
redistributor_18[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 18.
redistributor_19[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 19.
redistributor_2[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 2.
redistributor_20[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 20
redistributor_21[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 21.
redistributor_22[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 22.
redistributor_23[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 23
redistributor_24[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 24.
redistributor_25[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 25.
redistributor_26[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 26.
redistributor_27[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 27.
redistributor_28[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 28.
redistributor_29[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 29.
redistributor_3[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 3.
redistributor_30[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 30.
redistributor_31[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 31.
redistributor_4[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 4.
redistributor_5[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 5.
redistributor_6[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 6.

Name	Protocol	Type	Description
redistributor_7[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 7.
redistributor_8[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 8.
redistributor_9[8]	GICv3Comms	Master	Input from and output to CPU interface for cluster 9.
reset	Signal	Slave	Reset.
spi_in[988]	Signal	Slave	Shared peripheral interrupts.
wake_request_0[8]	Signal	Master	Power management outputs of cluster 0.
wake_request_1[8]	Signal	Master	Power management outputs of cluster 1.
wake_request_10[8]	Signal	Master	Power management outputs of cluster 10.
wake_request_11[8]	Signal	Master	Power management outputs of cluster 11.
wake_request_12[8]	Signal	Master	Power management outputs of cluster 12.
wake_request_13[8]	Signal	Master	Power management outputs of cluster 13.
wake_request_14[8]	Signal	Master	Power management outputs of cluster 14.
wake_request_15[8]	Signal	Master	Power management outputs of cluster 15.
wake_request_16[8]	Signal	Master	Power management outputs of cluster 16.
wake_request_17[8]	Signal	Master	Power management outputs of cluster 17.
wake_request_18[8]	Signal	Master	Power management outputs of cluster 18.
wake_request_19[8]	Signal	Master	Power management outputs of cluster 19.
wake_request_2[8]	Signal	Master	Power management outputs of cluster 2.
wake_request_20[8]	Signal	Master	Power management outputs of cluster 20.
wake_request_21[8]	Signal	Master	Power management outputs of cluster 21.
wake_request_22[8]	Signal	Master	Power management outputs of cluster 22.
wake_request_23[8]	Signal	Master	Power management outputs of cluster 23.
wake_request_24[8]	Signal	Master	Power management outputs of cluster 24.
wake_request_25[8]	Signal	Master	Power management outputs of cluster 25.
wake_request_26[8]	Signal	Master	Power management outputs of cluster 26.
wake_request_27[8]	Signal	Master	Power management outputs of cluster 27.
wake_request_28[8]	Signal	Master	Power management outputs of cluster 28.
wake_request_29[8]	Signal	Master	Power management outputs of cluster 29.
wake_request_3[8]	Signal	Master	Power management outputs of cluster 3.
wake_request_30[8]	Signal	Master	Power management outputs of cluster 30.
wake_request_31[8]	Signal	Master	Power management outputs of cluster 31.
wake_request_4[8]	Signal	Master	Power management outputs of cluster 4.
wake_request_5[8]	Signal	Master	Power management outputs of cluster 5.
wake_request_6[8]	Signal	Master	Power management outputs of cluster 6.
wake_request_7[8]	Signal	Master	Power management outputs of cluster 7.
wake_request_8[8]	Signal	Master	Power management outputs of cluster 8.
wake_request_9[8]	Signal	Master	Power management outputs of cluster 9.

Parameters for GIC500_ClusterPorts

GICD_ITARGETSR-RAZWI

If true, the GICD_ITARGETS registers are RAZ/WI

Type

bool

Default value

0x0

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

ITS-threaded-command-queue

Enable execution of ITS commands in a separate thread which is sometimes required for cosimulation

Type

bool

Default value

0x1

SPI-count

Number of SPIs that are implemented.

Type

int

Default value

0xe0

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

cpus_per_cluster_0

Number of cores within cluster 0

Type

int

Default value

0x1

cpus_per_cluster_1

Number of cores within cluster 1

Type

int

Default value

0x1

cpus_per_cluster_10

Number of cores within cluster 10

Type

int

Default value

0x1

cpus_per_cluster_11

Number of cores within cluster 11

Type

int

Default value

0x1

cpus_per_cluster_12

Number of cores within cluster 12

Type

int

Default value

0x1

cpus_per_cluster_13

Number of cores within cluster 13

Type

int

Default value

0x1

cpus_per_cluster_14

Number of cores within cluster 14

Type

int

Default value

0x1

cpus_per_cluster_15

Number of cores within cluster 15

Type

int

Default value

0x1

cpus_per_cluster_16

Number of cores within cluster 16

Type

int

Default value

0x1

cpus_per_cluster_17

Number of cores within cluster 17

Type

int

Default value

0x1

cpus_per_cluster_18

Number of cores within cluster 18

Type

int

Default value

0x1

cpus_per_cluster_19

Number of cores within cluster 19

Type

int

Default value

0x1

cpus_per_cluster_2

Number of cores within cluster 2

Type

int

Default value

0x1

cpus_per_cluster_20

Number of cores within cluster 20

Type

int

Default value

0x1

cpus_per_cluster_21

Number of cores within cluster 21

Type

int

Default value

0x1

cpus_per_cluster_22

Number of cores within cluster 22

Type

int

Default value

0x1

cpus_per_cluster_23

Number of cores within cluster 23

Type

int

Default value

0x1

cpus_per_cluster_24

Number of cores within cluster 24

Type

int

Default value

0x1

cpus_per_cluster_25

Number of cores within cluster 25

Type

int

Default value

0x1

cpus_per_cluster_26

Number of cores within cluster 26

Type

int

Default value

0x1

cpus_per_cluster_27

Number of cores within cluster 27

Type

int

Default value

0x1

cpus_per_cluster_28

Number of cores within cluster 28

Type

int

Default value

0x1

cpus_per_cluster_29

Number of cores within cluster 29

Type

int

Default value

0x1

cpus_per_cluster_3

Number of cores within cluster 3

Type

int

Default value

0x1

cpus_per_cluster_30

Number of cores within cluster 30

Type

int

Default value

0x1

cpus_per_cluster_31

Number of cores within cluster 31

Type

int

Default value

0x1

cpus_per_cluster_4

Number of cores within cluster 4

Type

int

Default value

0x1

cpus_per_cluster_5

Number of cores within cluster 5

Type

int

Default value

0x1

cpus_per_cluster_6

Number of cores within cluster 6

Type

int

Default value

0x1

cpus_per_cluster_7

Number of cores within cluster 7

Type

int

Default value

0x1

cpus_per_cluster_8

Number of cores within cluster 8

Type

int

Default value

0x1

cpus_per_cluster_9

Number of cores within cluster 9

Type

int

Default value

0x1

delay-ITS-accesses

Delay accesses from the ITS until GICR_SYNCRR is read.

Type

bool

Default value

0x1

delay-redistributor-accesses

Delay memory accesses from the redistributor until GICR_SYNCRR is read.

Type

bool

Default value

0x1

enable_protocol_checking

Enable/disable protocol checking at cpu interface

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

has-two-security-states

If true, has two security states

Type

bool

Default value

0x1

num_clusters

Number of implemented affinity level1 clusters

Type

int

Default value

0x1

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

redistributor-threaded-sync

Enable execution of redistributor delayed transactions in a separate thread which is sometimes required for cosimulation

Type

bool

Default value

0x1

reg-base

GIC500 base address

Type

int

Default value
0x10000000

using-generated-memorymap
Use the generated memorymap for the GIC500 and warn if superfluous parameters are passed

Type
bool

Default value
0x1

wakeup-on-reset
Go against specification and start redistributors in woken-up state at reset. This allows software that was written for previous versions of the GICv3 specification to work correctly. This should not be used for production code or when the distributor is used separately from the core fast model.

Type
bool

Default value
0x0

3.10.30 GIC500_Filter

GIC500 Component for distribution of interrupts, filtering version for validation. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-942: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC500_Filter

This model has the following Iris instances:

Table 3-943: GIC500_Filter Iris instances

InstanceName	ComponentName
GIC500_Filter	GIC_IRI
GIC500_Filter.ITS0	GICv3InterruptTranslationService
GIC500_Filter.rd_0	GICv3RedistributorInternal
GIC500_Filter.rd_0_0	GICv3RedistributorInternal

InstanceName	ComponentName
GIC500_Filter.rd_0_0_0	GICv3RedistributorInternal
GIC500_Filter.rd_0_0_0_0	GICv3Redistributor
GIC500_Filter.rd_t1	GICv3Distributor

This model has the following MTI trace components:

Table 3-944: GIC500_Filter MTI instances

InstanceName	ComponentName
GIC500_Filter	GICv3IRI
GIC500_Filter.ITS0	GICv3InterruptTranslationService
GIC500_Filter.rd_0	GICv3RedistributorInternal
GIC500_Filter.rd_0_0	GICv3RedistributorInternal
GIC500_Filter.rd_0_0_0	GICv3RedistributorInternal
GIC500_Filter.rd_0_0_0_0	GICv3Redistributor
GIC500_Filter.rd_t1	GICv3Distributor

GIC500_Filter contains the following CADI targets:

- GIC500

About GIC500_Filter

This is a single-component implementation of the GICv3 architecture with support for 256 cores. You can configure the model to support a maximum of 32 clusters with 8 cores per cluster. Use it with an Arm®v8-A core to deliver interrupts. It supports a single Interrupt Translation Service for message-based interrupts. It supports the architectural features, but does not support the **IMPLEMENTATION DEFINED** features.



Note

- Set the `FASTSIM_GIC_MEMORY_MAP` environment variable to 1 to print to `stderr` the memory map of any GICv3 or later models that are included in the platform being run.
- For writes to `GITS_TRANSLATE64R`, the `MSIRewriter` component can be used to create the correct data transactions based on the device ID. For more details about `GITS_TRANSLATE64R`, see [3.3.4 MSIRewriter](#) on page 139.

MasterID, ExtendedID, and UserFlags

This table shows how this model encodes information in the `MasterID`, `ExtendedID`, and `UserFlags` bus attributes:

Table 3-945: pvbuss port

PVBus attribute	Bits used	Property encoded	Notes
MasterID	Bits[31:0]	Master ID that invoked the register access	-
ExtendedID	-	-	Not used.

PVBus attribute	Bits used	Property encoded	Notes
UserFlags	-	-	Not used.



The pvbus_m port does not use MasterID, ExtendedID, Or UserFlags.

Ports for GIC500_Filter

Table 3-946: Ports

Name	Protocol	Type	Description
cfgsdisable	Signal	Slave	Disable some SPIs signal.
cpu_active_0[8]	Signal	Slave	cpu_active pins of cluster 0.
cpu_active_1[8]	Signal	Slave	cpu_active pins of cluster 1.
cpu_active_10[8]	Signal	Slave	cpu_active pins of cluster 10.
cpu_active_11[8]	Signal	Slave	cpu_active pins of cluster 11.
cpu_active_12[8]	Signal	Slave	cpu_active pins of cluster 12.
cpu_active_13[8]	Signal	Slave	cpu_active pins of cluster 13.
cpu_active_14[8]	Signal	Slave	cpu_active pins of cluster 14.
cpu_active_15[8]	Signal	Slave	cpu_active pins of cluster 15.
cpu_active_16[8]	Signal	Slave	cpu_active pins of cluster 16.
cpu_active_17[8]	Signal	Slave	cpu_active pins of cluster 17.
cpu_active_18[8]	Signal	Slave	cpu_active pins of cluster 18.
cpu_active_19[8]	Signal	Slave	cpu_active pins of cluster 19.
cpu_active_2[8]	Signal	Slave	cpu_active pins of cluster 2.
cpu_active_20[8]	Signal	Slave	cpu_active pins of cluster 20.
cpu_active_21[8]	Signal	Slave	cpu_active pins of cluster 21.
cpu_active_22[8]	Signal	Slave	cpu_active pins of cluster 22.
cpu_active_23[8]	Signal	Slave	cpu_active pins of cluster 23.
cpu_active_24[8]	Signal	Slave	cpu_active pins of cluster 24.
cpu_active_25[8]	Signal	Slave	cpu_active pins of cluster 25.
cpu_active_26[8]	Signal	Slave	cpu_active pins of cluster 26.
cpu_active_27[8]	Signal	Slave	cpu_active pins of cluster 27.
cpu_active_28[8]	Signal	Slave	cpu_active pins of cluster 28.
cpu_active_29[8]	Signal	Slave	cpu_active pins of cluster 29.
cpu_active_3[8]	Signal	Slave	cpu_active pins of cluster 3.
cpu_active_30[8]	Signal	Slave	cpu_active pins of cluster 30.
cpu_active_31[8]	Signal	Slave	cpu_active pins of cluster 31.
cpu_active_4[8]	Signal	Slave	cpu_active pins of cluster 4.
cpu_active_5[8]	Signal	Slave	cpu_active pins of cluster 5.

Name	Protocol	Type	Description
cpu_active_6[8]	Signal	Slave	cpu_active pins of cluster 6.
cpu_active_7[8]	Signal	Slave	cpu_active pins of cluster 7.
cpu_active_8[8]	Signal	Slave	cpu_active pins of cluster 8.
cpu_active_9[8]	Signal	Slave	cpu_active pins of cluster 9.
po_reset	Signal	Slave	Power on reset.
ppi16_in_0[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 0.
ppi16_in_1[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 1.
ppi16_in_10[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 10.
ppi16_in_11[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 11.
ppi16_in_12[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 12.
ppi16_in_13[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 13.
ppi16_in_14[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 14.
ppi16_in_15[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 15.
ppi16_in_16[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 16.
ppi16_in_17[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 17.
ppi16_in_18[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 18.
ppi16_in_19[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 19.
ppi16_in_2[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 2.
ppi16_in_20[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 20.
ppi16_in_21[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 21.
ppi16_in_22[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 22.
ppi16_in_23[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 23.
ppi16_in_24[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 24.
ppi16_in_25[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 25.
ppi16_in_26[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 26.
ppi16_in_27[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 27.
ppi16_in_28[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 28.
ppi16_in_29[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 29.
ppi16_in_3[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 3.
ppi16_in_30[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 30.
ppi16_in_31[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 31.
ppi16_in_4[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 4.
ppi16_in_5[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 5.
ppi16_in_6[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 6.
ppi16_in_7[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 7.
ppi16_in_8[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 8.
ppi16_in_9[8]	Signal	Slave	Private peripheral interrupts interrupts (ID16) of cluster 9.
ppi17_in_0[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 0.
ppi17_in_1[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 1.

Name	Protocol	Type	Description
ppi17_in_10[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 10.
ppi17_in_11[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 11.
ppi17_in_12[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 12.
ppi17_in_13[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 13.
ppi17_in_14[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 14.
ppi17_in_15[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 15.
ppi17_in_16[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 16.
ppi17_in_17[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 17.
ppi17_in_18[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 18.
ppi17_in_19[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 19.
ppi17_in_2[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 2.
ppi17_in_20[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 20.
ppi17_in_21[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 21.
ppi17_in_22[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 22.
ppi17_in_23[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 23.
ppi17_in_24[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 24.
ppi17_in_25[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 25.
ppi17_in_26[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 26.
ppi17_in_27[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 27.
ppi17_in_28[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 28.
ppi17_in_29[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 29.
ppi17_in_3[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 3.
ppi17_in_30[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 30.
ppi17_in_31[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 31.
ppi17_in_4[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 4.
ppi17_in_5[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 5.
ppi17_in_6[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 6.
ppi17_in_7[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 7.
ppi17_in_8[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 8.
ppi17_in_9[8]	Signal	Slave	Private peripheral interrupts interrupts (ID17) of cluster 9.
ppi18_in_0[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 0.
ppi18_in_1[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 1.
ppi18_in_10[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 10.
ppi18_in_11[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 11.
ppi18_in_12[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 12.
ppi18_in_13[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 13.
ppi18_in_14[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 14.
ppi18_in_15[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 15.
ppi18_in_16[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 16.

Name	Protocol	Type	Description
ppi18_in_17[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 17.
ppi18_in_18[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 18.
ppi18_in_19[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 19.
ppi18_in_2[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 2.
ppi18_in_20[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 20.
ppi18_in_21[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 21.
ppi18_in_22[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 22.
ppi18_in_23[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 23.
ppi18_in_24[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 24.
ppi18_in_25[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 25.
ppi18_in_26[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 26.
ppi18_in_27[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 27.
ppi18_in_28[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 28.
ppi18_in_29[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 29.
ppi18_in_3[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 3.
ppi18_in_30[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 30.
ppi18_in_31[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 31.
ppi18_in_4[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 4.
ppi18_in_5[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 5.
ppi18_in_6[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 6.
ppi18_in_7[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 7.
ppi18_in_8[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 8.
ppi18_in_9[8]	Signal	Slave	Private peripheral interrupts interrupts (ID18) of cluster 9.
ppi19_in_0[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 0.
ppi19_in_1[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 1.
ppi19_in_10[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 10.
ppi19_in_11[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 11.
ppi19_in_12[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 12.
ppi19_in_13[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 13.
ppi19_in_14[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 14.
ppi19_in_15[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 15.
ppi19_in_16[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 16.
ppi19_in_17[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 17.
ppi19_in_18[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 18.
ppi19_in_19[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 19.
ppi19_in_2[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 2.
ppi19_in_20[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 20.
ppi19_in_21[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 21.
ppi19_in_22[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 22.

Name	Protocol	Type	Description
ppi19_in_23[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 23.
ppi19_in_24[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 24.
ppi19_in_25[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 25.
ppi19_in_26[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 26.
ppi19_in_27[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 27.
ppi19_in_28[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 28.
ppi19_in_29[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 29.
ppi19_in_3[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 3.
ppi19_in_30[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 30.
ppi19_in_31[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 31.
ppi19_in_4[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 4.
ppi19_in_5[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 5.
ppi19_in_6[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 6.
ppi19_in_7[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 7.
ppi19_in_8[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 8.
ppi19_in_9[8]	Signal	Slave	Private peripheral interrupts (ID19) of cluster 9.
ppi20_in_0[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 0.
ppi20_in_1[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 1.
ppi20_in_10[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 10.
ppi20_in_11[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 11.
ppi20_in_12[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 12.
ppi20_in_13[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 13.
ppi20_in_14[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 14.
ppi20_in_15[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 15.
ppi20_in_16[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 16.
ppi20_in_17[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 17.
ppi20_in_18[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 18.
ppi20_in_19[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 19.
ppi20_in_2[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 2.
ppi20_in_20[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 20.
ppi20_in_21[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 21.
ppi20_in_22[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 22.
ppi20_in_23[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 23.
ppi20_in_24[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 24.
ppi20_in_25[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 25.
ppi20_in_26[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 26.
ppi20_in_27[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 27.
ppi20_in_28[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 28.
ppi20_in_29[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 29.

Name	Protocol	Type	Description
ppi20_in_3[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 3.
ppi20_in_30[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 30.
ppi20_in_31[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 31.
ppi20_in_4[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 4.
ppi20_in_5[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 5.
ppi20_in_6[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 6.
ppi20_in_7[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 7.
ppi20_in_8[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 8.
ppi20_in_9[8]	Signal	Slave	Private peripheral interrupts (ID20) of cluster 9.
ppi21_in_0[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 0.
ppi21_in_1[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 1.
ppi21_in_10[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 10.
ppi21_in_11[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 11.
ppi21_in_12[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 12.
ppi21_in_13[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 13.
ppi21_in_14[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 14.
ppi21_in_15[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 15.
ppi21_in_16[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 16.
ppi21_in_17[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 17.
ppi21_in_18[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 18.
ppi21_in_19[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 19.
ppi21_in_2[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 2.
ppi21_in_20[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 20.
ppi21_in_21[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 21.
ppi21_in_22[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 22.
ppi21_in_23[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 23.
ppi21_in_24[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 24.
ppi21_in_25[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 25.
ppi21_in_26[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 26.
ppi21_in_27[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 27.
ppi21_in_28[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 28.
ppi21_in_29[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 29.
ppi21_in_3[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 3.
ppi21_in_30[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 30.
ppi21_in_31[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 31.
ppi21_in_4[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 4.
ppi21_in_5[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 5.
ppi21_in_6[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 6.
ppi21_in_7[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 7.

Name	Protocol	Type	Description
ppi21_in_8[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 8.
ppi21_in_9[8]	Signal	Slave	Private peripheral interrupts (ID21) of cluster 9.
ppi22_in_0[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 0.
ppi22_in_1[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 1.
ppi22_in_10[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 10.
ppi22_in_11[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 11.
ppi22_in_12[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 12.
ppi22_in_13[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 13.
ppi22_in_14[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 14.
ppi22_in_15[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 15.
ppi22_in_16[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 16.
ppi22_in_17[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 17.
ppi22_in_18[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 18.
ppi22_in_19[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 19.
ppi22_in_2[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 2.
ppi22_in_20[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 20.
ppi22_in_21[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 21.
ppi22_in_22[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 22.
ppi22_in_23[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 23.
ppi22_in_24[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 24.
ppi22_in_25[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 25.
ppi22_in_26[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 26.
ppi22_in_27[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 27.
ppi22_in_28[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 28.

Name	Protocol	Type	Description
ppi22_in_29[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 29.
ppi22_in_3[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 3.
ppi22_in_30[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 30.
ppi22_in_31[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 31.
ppi22_in_4[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 4.
ppi22_in_5[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 5.
ppi22_in_6[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 6.
ppi22_in_7[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 7.
ppi22_in_8[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 8.
ppi22_in_9[8]	Signal	Slave	Private peripheral interrupts (ID22, typically the Debug Communications Channel (DCC) interrupt) of cluster 9.
ppi23_in_0[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 0.
ppi23_in_1[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 1.
ppi23_in_10[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 10.
ppi23_in_11[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 11.
ppi23_in_12[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 12.
ppi23_in_13[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 13.
ppi23_in_14[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 14.
ppi23_in_15[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 15.
ppi23_in_16[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 16.
ppi23_in_17[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 17.
ppi23_in_18[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 18.
ppi23_in_19[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 19.
ppi23_in_2[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 2.

Name	Protocol	Type	Description
ppi23_in_20[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 20.
ppi23_in_21[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 21.
ppi23_in_22[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 22.
ppi23_in_23[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 23.
ppi23_in_24[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 24.
ppi23_in_25[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 25.
ppi23_in_26[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 26.
ppi23_in_27[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 27.
ppi23_in_28[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 28.
ppi23_in_29[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 29.
ppi23_in_3[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 3.
ppi23_in_30[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 30.
ppi23_in_31[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 31.
ppi23_in_4[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 4.
ppi23_in_5[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 5.
ppi23_in_6[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 6.
ppi23_in_7[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 7.
ppi23_in_8[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 8.
ppi23_in_9[8]	Signal	Slave	Private peripheral interrupts (ID23, typically the Performance Counter (PMU) overflow interrupt) of cluster 9.
ppi24_in_0[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 0.
ppi24_in_1[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 1.
ppi24_in_10[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 10.
ppi24_in_11[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 11.

Name	Protocol	Type	Description
ppi24_in_12[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 12.
ppi24_in_13[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 13.
ppi24_in_14[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 14.
ppi24_in_15[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 15.
ppi24_in_16[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 16.
ppi24_in_17[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 17.
ppi24_in_18[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 18.
ppi24_in_19[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 19.
ppi24_in_2[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 2.
ppi24_in_20[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 20.
ppi24_in_21[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 21.
ppi24_in_22[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 22.
ppi24_in_23[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 23.
ppi24_in_24[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 24.
ppi24_in_25[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 25.
ppi24_in_26[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 26.
ppi24_in_27[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 27.
ppi24_in_28[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 28.
ppi24_in_29[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 29.
ppi24_in_3[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 3.
ppi24_in_30[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 30.
ppi24_in_31[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 31.
ppi24_in_4[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 4.

Name	Protocol	Type	Description
ppi24_in_5[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 5.
ppi24_in_6[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 6.
ppi24_in_7[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 7.
ppi24_in_8[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 8.
ppi24_in_9[8]	Signal	Slave	Private peripheral interrupts (ID24, typically the Cross Trigger Interface (CTI) interrupt) of cluster 9.
ppi25_in_0[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 0.
ppi25_in_1[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 1.
ppi25_in_10[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 10.
ppi25_in_11[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 11.
ppi25_in_12[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 12.
ppi25_in_13[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 13.
ppi25_in_14[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 14.
ppi25_in_15[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 15.
ppi25_in_16[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 16.
ppi25_in_17[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 17.
ppi25_in_18[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 18.
ppi25_in_19[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 19.
ppi25_in_2[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 2.
ppi25_in_20[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 20.
ppi25_in_21[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 21.
ppi25_in_22[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 22.
ppi25_in_23[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 23.
ppi25_in_24[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 24.

Name	Protocol	Type	Description
ppi25_in_25[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 25.
ppi25_in_26[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 26.
ppi25_in_27[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 27.
ppi25_in_28[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 28.
ppi25_in_29[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 29.
ppi25_in_3[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 3.
ppi25_in_30[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 30.
ppi25_in_31[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 31.
ppi25_in_4[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 4.
ppi25_in_5[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 5.
ppi25_in_6[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 6.
ppi25_in_7[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 7.
ppi25_in_8[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 8.
ppi25_in_9[8]	Signal	Slave	Private peripheral interrupts (ID25, typically the virtual CPU interface maintenance interrupt) of cluster 9.
ppi26_in_0[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 0.
ppi26_in_1[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 1.
ppi26_in_10[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 10.
ppi26_in_11[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 11.
ppi26_in_12[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 12.
ppi26_in_13[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 13.
ppi26_in_14[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 14.
ppi26_in_15[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 15.
ppi26_in_16[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 16.
ppi26_in_17[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 17.
ppi26_in_18[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 18.
ppi26_in_19[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 19.
ppi26_in_2[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 2.
ppi26_in_20[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 20.
ppi26_in_21[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 21.

Name	Protocol	Type	Description
ppi26_in_22[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 22.
ppi26_in_23[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 23.
ppi26_in_24[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 24.
ppi26_in_25[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 25.
ppi26_in_26[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 26.
ppi26_in_27[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 27.
ppi26_in_28[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 28.
ppi26_in_29[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 29.
ppi26_in_3[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 3.
ppi26_in_30[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 30.
ppi26_in_31[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 31.
ppi26_in_4[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 4.
ppi26_in_5[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 5.
ppi26_in_6[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 6.
ppi26_in_7[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 7.
ppi26_in_8[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 8.
ppi26_in_9[8]	Signal	Slave	Private peripheral interrupts (ID26, typically the hypervisor timer) of cluster 9.
ppi27_in_0[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 0.
ppi27_in_1[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 1.
ppi27_in_10[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 10.
ppi27_in_11[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 11.
ppi27_in_12[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 12.
ppi27_in_13[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 13.
ppi27_in_14[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 14.
ppi27_in_15[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 15.
ppi27_in_16[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 16.
ppi27_in_17[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 17.
ppi27_in_18[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 18.
ppi27_in_19[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 19.
ppi27_in_2[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 2.
ppi27_in_20[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 20.
ppi27_in_21[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 21.
ppi27_in_22[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 22.
ppi27_in_23[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 23.
ppi27_in_24[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 24.
ppi27_in_25[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 25.
ppi27_in_26[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 26.
ppi27_in_27[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 27.
ppi27_in_28[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 28.

Name	Protocol	Type	Description
ppi27_in_29[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 29.
ppi27_in_3[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 3.
ppi27_in_30[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 30.
ppi27_in_31[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 31.
ppi27_in_4[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 4.
ppi27_in_5[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 5.
ppi27_in_6[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 6.
ppi27_in_7[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 7.
ppi27_in_8[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 8.
ppi27_in_9[8]	Signal	Slave	Private peripheral interrupts (ID27, typically the virtual timer) of cluster 9.
ppi28_in_0[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 0.
ppi28_in_1[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 1.
ppi28_in_10[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 10.
ppi28_in_11[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 11.
ppi28_in_12[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 12.
ppi28_in_13[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 13.
ppi28_in_14[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 14.
ppi28_in_15[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 15.
ppi28_in_16[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 16.
ppi28_in_17[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 17.
ppi28_in_18[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 18.
ppi28_in_19[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 19.
ppi28_in_2[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 2.
ppi28_in_20[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 20.
ppi28_in_21[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 21.
ppi28_in_22[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 22.
ppi28_in_23[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 23.
ppi28_in_24[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 24.
ppi28_in_25[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 25.
ppi28_in_26[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 26.
ppi28_in_27[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 27.
ppi28_in_28[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 28.
ppi28_in_29[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 29.
ppi28_in_3[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 3.
ppi28_in_30[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 30.
ppi28_in_31[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 31.
ppi28_in_4[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 4.
ppi28_in_5[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 5.
ppi28_in_6[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 6.

Name	Protocol	Type	Description
ppi28_in_7[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 7.
ppi28_in_8[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 8.
ppi28_in_9[8]	Signal	Slave	Private peripheral interrupts (ID28) of cluster 9.
ppi29_in_0[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 0.
ppi29_in_1[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 1.
ppi29_in_10[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 10.
ppi29_in_11[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 11.
ppi29_in_12[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 12.
ppi29_in_13[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 13.
ppi29_in_14[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 14.
ppi29_in_15[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 15.
ppi29_in_16[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 16.
ppi29_in_17[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 17.
ppi29_in_18[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 18.
ppi29_in_19[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 19.
ppi29_in_2[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 2.
ppi29_in_20[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 20.
ppi29_in_21[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 21.
ppi29_in_22[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 22.
ppi29_in_23[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 23.
ppi29_in_24[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 24.
ppi29_in_25[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 25.
ppi29_in_26[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 26.
ppi29_in_27[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 27.
ppi29_in_28[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 28.
ppi29_in_29[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 29.
ppi29_in_3[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 3.
ppi29_in_30[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 30.
ppi29_in_31[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 31.
ppi29_in_4[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 4.
ppi29_in_5[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 5.
ppi29_in_6[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 6.
ppi29_in_7[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 7.
ppi29_in_8[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 8.
ppi29_in_9[8]	Signal	Slave	Private peripheral interrupts (ID29, typically the Secure physical timer) of cluster 9.
ppi30_in_0[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 0.
ppi30_in_1[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 1.

Name	Protocol	Type	Description
ppi30_in_10[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 10.
ppi30_in_11[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 11.
ppi30_in_12[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 12.
ppi30_in_13[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 13.
ppi30_in_14[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 14.
ppi30_in_15[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 15.
ppi30_in_16[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 16.
ppi30_in_17[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 17.
ppi30_in_18[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 18.
ppi30_in_19[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 19.
ppi30_in_2[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 2.
ppi30_in_20[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 20.
ppi30_in_21[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 21.
ppi30_in_22[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 22.
ppi30_in_23[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 23.
ppi30_in_24[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 24.
ppi30_in_25[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 25.
ppi30_in_26[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 26.
ppi30_in_27[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 27.
ppi30_in_28[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 28.
ppi30_in_29[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 29.
ppi30_in_3[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 3.
ppi30_in_30[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 30.

Name	Protocol	Type	Description
ppi30_in_31[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 31.
ppi30_in_4[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 4.
ppi30_in_5[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 5.
ppi30_in_6[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 6.
ppi30_in_7[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 7.
ppi30_in_8[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 8.
ppi30_in_9[8]	Signal	Slave	Private peripheral interrupts (ID30, typically the Non-secure physical timer) of cluster 9.
ppi31_in_0[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 0.
ppi31_in_1[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 1.
ppi31_in_10[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 10.
ppi31_in_11[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 11.
ppi31_in_12[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 12.
ppi31_in_13[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 13.
ppi31_in_14[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 14.
ppi31_in_15[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 15.
ppi31_in_16[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 16.
ppi31_in_17[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 17.
ppi31_in_18[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 18.
ppi31_in_19[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 19.
ppi31_in_2[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 2.
ppi31_in_20[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 20.
ppi31_in_21[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 21.
ppi31_in_22[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 22.
ppi31_in_23[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 23.
ppi31_in_24[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 24.
ppi31_in_25[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 25.
ppi31_in_26[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 26.
ppi31_in_27[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 27.
ppi31_in_28[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 28.
ppi31_in_29[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 29.
ppi31_in_3[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 3.
ppi31_in_30[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 30.
ppi31_in_31[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 31.
ppi31_in_4[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 4.

Name	Protocol	Type	Description
ppi31_in_5[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 5.
ppi31_in_6[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 6.
ppi31_in_7[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 7.
ppi31_in_8[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 8.
ppi31_in_9[8]	Signal	Slave	Private peripheral interrupts (ID31) of cluster 9.
pvbuss_filtermiss_m	PVBus	Master	passthrough for transactions not targetting one of the pages associated with the IRI.
pvbuss_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbuss_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Reset.
spi_in[988]	Signal	Slave	Shared peripheral interrupts.
wake_request_0[8]	Signal	Master	Power management outputs of cluster 0.
wake_request_1[8]	Signal	Master	Power management outputs of cluster 1.
wake_request_10[8]	Signal	Master	Power management outputs of cluster 10.
wake_request_11[8]	Signal	Master	Power management outputs of cluster 11.
wake_request_12[8]	Signal	Master	Power management outputs of cluster 12.
wake_request_13[8]	Signal	Master	Power management outputs of cluster 13.
wake_request_14[8]	Signal	Master	Power management outputs of cluster 14.
wake_request_15[8]	Signal	Master	Power management outputs of cluster 15.
wake_request_16[8]	Signal	Master	Power management outputs of cluster 16.
wake_request_17[8]	Signal	Master	Power management outputs of cluster 17.
wake_request_18[8]	Signal	Master	Power management outputs of cluster 18.
wake_request_19[8]	Signal	Master	Power management outputs of cluster 19.
wake_request_2[8]	Signal	Master	Power management outputs of cluster 2.
wake_request_20[8]	Signal	Master	Power management outputs of cluster 20.
wake_request_21[8]	Signal	Master	Power management outputs of cluster 21.
wake_request_22[8]	Signal	Master	Power management outputs of cluster 22.
wake_request_23[8]	Signal	Master	Power management outputs of cluster 23.
wake_request_24[8]	Signal	Master	Power management outputs of cluster 24.
wake_request_25[8]	Signal	Master	Power management outputs of cluster 25.
wake_request_26[8]	Signal	Master	Power management outputs of cluster 26.
wake_request_27[8]	Signal	Master	Power management outputs of cluster 27.
wake_request_28[8]	Signal	Master	Power management outputs of cluster 28.
wake_request_29[8]	Signal	Master	Power management outputs of cluster 29.
wake_request_3[8]	Signal	Master	Power management outputs of cluster 3.
wake_request_30[8]	Signal	Master	Power management outputs of cluster 30.
wake_request_31[8]	Signal	Master	Power management outputs of cluster 31.
wake_request_4[8]	Signal	Master	Power management outputs of cluster 4.
wake_request_5[8]	Signal	Master	Power management outputs of cluster 5.

Name	Protocol	Type	Description
wake_request_6[8]	Signal	Master	Power management outputs of cluster 6.
wake_request_7[8]	Signal	Master	Power management outputs of cluster 7.
wake_request_8[8]	Signal	Master	Power management outputs of cluster 8.
wake_request_9[8]	Signal	Master	Power management outputs of cluster 9.

Parameters for GIC500_Filter

GICD_ITARGETSR-RAZWI

If true, the GICD_ITARGETS registers are RAZ/WI

Type

bool

Default value

0x0

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

ITS-threaded-command-queue

Enable execution of ITS commands in a separate thread which is sometimes required for cosimulation

Type

bool

Default value

0x1

SPI-count

Number of SPIs that are implemented.

Type

int

Default value

0xe0

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

cpus_per_cluster_0

Number of cores within cluster 0

Type

int

Default value

0x1

cpus_per_cluster_1

Number of cores within cluster 1

Type

int

Default value

0x1

cpus_per_cluster_10

Number of cores within cluster 10

Type

int

Default value

0x1

cpus_per_cluster_11

Number of cores within cluster 11

Type

int

Default value

0x1

cpus_per_cluster_12

Number of cores within cluster 12

Type

int

Default value

0x1

cpus_per_cluster_13

Number of cores within cluster 13

Type

int

Default value

0x1

cpus_per_cluster_14

Number of cores within cluster 14

Type

int

Default value

0x1

cpus_per_cluster_15

Number of cores within cluster 15

Type

int

Default value

0x1

cpus_per_cluster_16

Number of cores within cluster 16

Type

int

Default value

0x1

cpus_per_cluster_17

Number of cores within cluster 17

Type

int

Default value

0x1

cpus_per_cluster_18

Number of cores within cluster 18

Type

int

Default value

0x1

cpus_per_cluster_19

Number of cores within cluster 19

Type

int

Default value

0x1

cpus_per_cluster_2

Number of cores within cluster 2

Type

int

Default value

0x1

cpus_per_cluster_20

Number of cores within cluster 20

Type

int

Default value

0x1

cpus_per_cluster_21

Number of cores within cluster 21

Type

int

Default value

0x1

cpus_per_cluster_22

Number of cores within cluster 22

Type

int

Default value

0x1

cpus_per_cluster_23

Number of cores within cluster 23

Type

int

Default value

0x1

cpus_per_cluster_24

Number of cores within cluster 24

Type

int

Default value

0x1

cpus_per_cluster_25

Number of cores within cluster 25

Type

int

Default value

0x1

cpus_per_cluster_26

Number of cores within cluster 26

Type

int

Default value

0x1

cpus_per_cluster_27

Number of cores within cluster 27

Type

int

Default value

0x1

cpus_per_cluster_28

Number of cores within cluster 28

Type

int

Default value

0x1

cpus_per_cluster_29

Number of cores within cluster 29

Type

int

Default value

0x1

cpus_per_cluster_3

Number of cores within cluster 3

Type

int

Default value

0x1

cpus_per_cluster_30

Number of cores within cluster 30

Type

int

Default value

0x1

cpus_per_cluster_31

Number of cores within cluster 31

Type

int

Default value

0x1

cpus_per_cluster_4

Number of cores within cluster 4

Type

int

Default value

0x1

cpus_per_cluster_5

Number of cores within cluster 5

Type

int

Default value

0x1

cpus_per_cluster_6

Number of cores within cluster 6

Type

int

Default value

0x1

cpus_per_cluster_7

Number of cores within cluster 7

Type

int

Default value

0x1

cpus_per_cluster_8

Number of cores within cluster 8

Type

int

Default value

0x1

cpus_per_cluster_9

Number of cores within cluster 9

Type

int

Default value

0x1

delay-ITS-accesses

Delay accesses from the ITS until GICR_SYNCR is read.

Type

bool

Default value

0x1

delay-redistributor-accesses

Delay memory accesses from the redistributor until GICR_SYNCR is read.

Type

bool

Default value

0x1

enable_protocol_checking

Enable/disable protocol checking at cpu interface

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

has-two-security-states

If true, has two security states

Type

bool

Default value

0x1

num_clusters

Number of implemented affinity level1 clusters

Type

int

Default value

0x1

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

redistributor-threaded-sync

Enable execution of redistributor delayed transactions in a separate thread which is sometimes required for cosimulation

Type

bool

Default value

0x1

reg-base

GIC500 base address

Type

int

Default value

0x10000000

using-generated-memorymap

Use the generated memorymap for the GIC500 and warn if superfluous parameters are passed

Type

bool

Default value

0x1

wakeup-on-reset

Go against specification and start redistributors in woken-up state at reset. This allows software that was written for previous versions of the GICv3 specification to work correctly. This should not be used for production code or when the distributor is used separately from the core fast model.

Type

bool

Default value

0x0

3.10.31 GIC600

GIC-600 IRI implementation: Single chip validation/filter component variant limited to 265 PE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-947: IP revisions support

Revision	Quality level
r1p6	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC600

This model has the following Iris instances:

Table 3-948: GIC600 Iris instances

InstanceName	ComponentName
GIC600	GIC_IRI
GIC600.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC600.ITS0	GICv3InterruptTranslationService
GIC600.rd_0	GICv3RedistributorInternal
GIC600.rd_0_0	GICv3RedistributorInternal
GIC600.rd_0_0_0	GICv3RedistributorInternal
GIC600.rd_0_0_0_0	GICv3Redistributor
GIC600.rd_t1	GICv3Distributor

This model has the following MTI trace components:

Table 3-949: GIC600 MTI instances

InstanceName	ComponentName
GIC600	GICv3IRI
GIC600.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC600.ITS0	GICv3InterruptTranslationService
GIC600.rd_0	GICv3RedistributorInternal
GIC600.rd_0_0	GICv3RedistributorInternal
GIC600.rd_0_0_0	GICv3RedistributorInternal
GIC600.rd_0_0_0_0	GICv3Redistributor
GIC600.rd_t1	GICv3Distributor

GIC600 contains the following CADI targets:

- GIC600

About GIC600

GIC600 and GIC600_Filter are minimal models of an Arm® GIC-600 Generic Interrupt Controller, suitable for single-chip or multichip systems, where multichip operation has been tested, with the following exceptions:

- LPI/LPI command is implemented but untested.
- Power operation is unimplemented.

They provide a simple configuration interface that allows designers to introduce GIC600-like functionality to their systems, while only implementing the architectural behavior, as defined by the GICv3 architecture.

All implementation-specific registers and functionality are implemented.

As with the other GIC components, there are two variants of the model with slightly different memory interfaces. Both GIC600 and GIC600_Filter have a `pvbuss` port for register accesses and

a `pvbuss_m` port for the LPI-related traffic from redistributors and the ITS. The `pvbuss_m` port is also used to compose multichip operation.

In addition, the GIC600_Filter variant has a `pvbuss_filtermiss_m` port, to which any transactions on the `pvbuss_s` port and not directed to a 4K page used by the GIC are forwarded unmodified. Such transactions are terminated in the component when using the GIC600 variant.

It is recommended to use the GIC600 variant in most cases.

Also, the RAS register is implemented and various RAS errors are reported through status registers. Not all the RAS errors are available because Fast Models does not model errors that can cause ECC errors. Fast Models supports error record 0 and error record 13+.



- Set the `FASTSIM_GIC_MEMORY_MAP` environment variable to 1 to print to `stderr` the memory map of any GICv3 or later models that are included in the platform being run.
- For writes to `GITS_TRANSLATE64R`, the MSIRewriter component can be used to create the correct data transactions based on the device ID. For more details about `GITS_TRANSLATE64R`, see [3.3.4 MSIRewriter](#) on page 139.

MasterID, ExtendedID, and UserFlags

This table shows how this model encodes information in the `MasterID`, `ExtendedID`, and `UserFlags` bus attributes:

Table 3-950: `pvbuss_s` port

PVBus attribute	Bits used	Property encoded	Notes
MasterID	Bits[31:0]	Master ID that invoked the register access	-
ExtendedID	-	-	Not used.
UserFlags	-	-	Not used.



The `pvbuss_m` port does not use `MasterID`, `ExtendedID`, or `UserFlags`.

Ports for GIC600

Table 3-951: Ports

Name	Protocol	Type	Description
<code>chip_id</code>	Value	Slave	<code>chip_id</code> port which is valid from GIC600 r1p2. Writing to this port for prior GIC600 version will be ignored.
<code>cpu_active_s[256]</code>	Signal	Slave	CPUActive pins.
<code>po_reset</code>	Signal	Slave	Resets.
<code>ppi_in_<n>[16]</code>	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.

Name	Protocol	Type	Description
pvbus_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Resets.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
wake_request[256]	Signal	Master	Power management outputs.

Parameters for GIC600

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value

DS-behaviour

GICD_CTLR.DS field behaviour: 0:RAZ/WI, 1:RAO/WI, 2:RW

Type

int

Default value

0x2

IIDR

GICD_IIDR and GICR_IIDR value. When the parameter is not given or is set to zero, the GIC600 model is configured to have the latest revision.

Type

int

Default value

0x0

ITS-ID-bits

Number of interrupt bits supported by ITS.

Type

int

Default value

0x10

ITS-collection-ID-bits

Number of collection bits supported by ITS (optional parameter, 0 => 16bits support and GITS_TYPER.CIL=0)

Type

int

Default value

0x8

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

PPI-count

Selects the number of PPI available for each PE as 8(id22-27,29,30), 12(id 20-31) or 16(id 16-31)

Type

int

Default value

0x10

RAS-CFI-support

If true, fault handling interrupt for corrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-FI-support

If true, fault handling interrupt is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UE-support

If true, In-band uncorrected error reporting is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UI-support

If true, error recovery interrupt for uncorrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

SPI-blocks

Number of SPI blocks supported by the IRI, each block contains 32 SPIs

Type

int

Default value

0x1e

affinity-width

A dotted quad indicating the bitwidth of fields at each affinity level

Type

string

Default value

4.8.8.8

chip-id

Chip ID when multichip operation is enabled.

Type

int

Default value

0x0

chip-select-affinity-level

Affinity level 2 or 3 can be used for chip select

Type

int

Default value

0x3

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

direct-lpi-support

Enable support for LPI operations through GICR registers

Type

bool

Default value

0x0

enable-multichip-operation

Enables multi-chip operation between Distributors in distributed GIC IRI

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

gict-allow-ns-reset

If true, non-secure read/write access to GICT register is allowed at reset. Not allowed otherwise. This emulates gict_allow_ns tie-off signal.

Type

bool

Default value

0x1

max-cores-supported-by-GCI

GCI can support 1-64 cores. Maximum supported cores in GCI is configurable by device

Type

int

Default value

0x8

max-pe-on-chip

Maximum number of cores on any single chip. This will be used to identify the target chip and core when multichip operation is enabled.

Type

int

Default value

0x4

multichip-threaded-dgi

Enable sending multichip DGI messages in a separate thread. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x1

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

redistributor-group

Redistributor grouping information with affinity as JSON : {"0":["0.0.0.0","0.0.0.1"],"1":["0.0.1.0","0.0.1.1"]} where RD with quad 0.0.0.0 and 0.0.0.1 belongs to RD group 0. All the RDs belong to one group when this parameter is not given.

Type

string

Default value**redistributor-group-file**

File path to redistributor grouping information with affinity as JSON. The file uses the same format as "redistributor-group" parameter.

Type

string

Default value**redistributor-power-managed-by-pwrr**

GIC600 dedistributor power management is done by updating GICR_PWRR register

Type

bool

Default value

0x1

reg-base

GIC-600 base address

Type

int

Default value

0x2c010000

reg-base-per-redistributor

Base address for each redistributor in the form '0.0.0.0=0x2c010000, 0.0.0.1=0x2c020000'. All redistributors must be specified and this overrides the reg-base parameter (except that reg-base will still be used for the top-level redistributor). If reg-base-per-redistributor-file is specified, this parameter is ignored.

Type

string

Default value

3.10.32 GIC600AE

GIC-600AE IRI implementation: Single chip validation/filter component variant limited to 265 PE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-952: IP revisions support

Revision	Quality level
r0p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC600AE

This model has the following Iris instances:

Table 3-953: GIC600AE Iris instances

InstanceName	ComponentName
GIC600AE	GIC_IRI
GIC600AE.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC600AE.ITS0	GICv3InterruptTranslationService
GIC600AE.rd_0	GICv3RedistributorInternal
GIC600AE.rd_0_0	GICv3RedistributorInternal
GIC600AE.rd_0_0_0	GICv3RedistributorInternal
GIC600AE.rd_0_0_0_0	GICv3Redistributor
GIC600AE.rd_t1	GICv3Distributor

This model has the following MTI trace components:

Table 3-954: GIC600AE MTI instances

InstanceName	ComponentName
GIC600AE	GICv3IRI
GIC600AE.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC600AE.ITS0	GICv3InterruptTranslationService
GIC600AE.rd_0	GICv3RedistributorInternal
GIC600AE.rd_0_0	GICv3RedistributorInternal
GIC600AE.rd_0_0_0	GICv3RedistributorInternal
GIC600AE.rd_0_0_0_0	GICv3Redistributor
GIC600AE.rd_t1	GICv3Distributor

GIC600AE contains the following CADI targets:

- GIC600AE

About GIC600AE

GIC600AE and GIC600AE_Filter are minimal models of an Arm® GIC-600AE Generic Interrupt Controller, suitable for single-chip or multichip systems, where multichip operation has been tested, with the following exceptions:

- LPI/LPI command is implemented but untested.
- Power operation is unimplemented.

They provide a simple configuration interface that allows designers to introduce GIC600AE-like functionality to their systems, while only implementing the architectural behavior, as defined by the GICv3 architecture.

All implementation-specific registers and functionality are implemented.

As with the other GIC components, there are two variants of the model with slightly different memory interfaces. Both GIC600AE and GIC600AE_Filter have a `pvbus_s` port for register accesses and a `pvbus_m` port for the LPI-related traffic from redistributors and the ITS. The `pvbus_m` port is also used to compose multichip operation.

In addition, the GIC600AE_Filter variant has a `pvbus_filtermiss_m` port, to which any transactions on the `pvbus_s` port and not directed to a 4K page used by the GIC are forwarded unmodified. Such transactions are terminated in the component when using the GIC600AE variant.

It is recommended to use the GIC600AE variant in most cases.

Also, the RAS register is implemented and various RAS errors are reported through status registers. Not all the RAS errors are available because Fast Models does not model errors that can cause ECC errors. Fast Models supports error record 0 and error record 13+. FMU registers are also implemented and are accessible by software.



- Set the `FASTSIM_GIC_MEMORY_MAP` environment variable to 1 to print to `stderr` the memory map of any GICv3 or later models that are included in the platform being run.
- For writes to `GITS_TRANSLATE64R`, the `MSIRewriter` component can be used to create the correct data transactions based on the device ID. For more details about `GITS_TRANSLATE64R`, see [3.3.4 MSIRewriter](#) on page 139.

MasterID, ExtendedID, and UserFlags

This table shows how this model encodes information in the `MasterID`, `ExtendedID`, and `UserFlags` bus attributes:

Table 3-955: `pvbus_s` port

PVBus attribute	Bits used	Property encoded	Notes
<code>MasterID</code>	Bits[31:0]	Master ID that invoked the register access	-
<code>ExtendedID</code>	-	-	Not used.
<code>UserFlags</code>	-	-	Not used.



The `pvbus_m` port does not use `MasterID`, `ExtendedID`, or `UserFlags`.

AE-specific features implemented

GIC600AE is a Functional Safety (FuSa) variant of GIC600. It has the following differences from GIC600:

- Both GIC600AE and GIC600 support RAS, but only GIC600AE supports the Fault Management Unit (FMU). This support is controlled by the `has-fmu` parameter which is true by default. The FMU resides in the Distributor and processes faults that are detected by the Safety Mechanisms from all blocks.
- In GIC600AE, the GIC reset pin is connected to FMU reset. Only reset changes the error record registers and FMU_ERRGSR to their reset values.
- In GIC600AE, the Safety Mechanism detects faults and forwards them to the FMU. The FMU forwards all errors to the Safety Island.
- The APB port has been added to GIC600AE for FuSa purposes. It does not exist on the GIC600.
- GIC600AE supports both Error Recovery Interrupt (ERI) and Fault Handling Interrupt (FHI).
- GIC600AE has the limitation that it only supports error injection through the FMU_SMINJERR register.

Ports for GIC600AE

Table 3-956: Ports

Name	Protocol	Type	Description
apb_bus	PVBus	Slave	APB4 port to FMU, which will be connected from safety island for FuSa
chip_id	Value	Slave	chip_id port used for multichip operation
cpu_active_s[256]	Signal	Slave	CPUActive pins.
fmu_error_int	Signal	Master	FuSa FMU error interrupt signal
fmu_fault_int	Signal	Master	FuSa FMU fault interrupt signal
po_reset	Signal	Slave	Resets. This is used as a dbg_reset in TRM, and also be used for cold reset for PMU and FMU.
ppi_in_<n>[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
pvbus_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Resets.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
wake_request[256]	Signal	Master	Power management outputs.

Parameters for GIC600AE

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value**DS-behaviour**

GICD_CTLR.DS field behaviour: 0:RAZ/WI, 1:RAO/WI, 2:RW

Type

int

Default value

0x2

IIDR

GICD_IIDR and GICR_IIDR value. When the parameter is not given or is set to zero, the GIC600 model is configured to have the latest revision.

Type

int

Default value

0x0

ITS-ID-bits

Number of interrupt bits supported by ITS.

Type

int

Default value

0x10

ITS-collection-ID-bits

Number of collection bits supported by ITS (optional parameter, 0 => 16bits support and GITS_TYPER.CIL=0)

Type

int

Default value

0x8

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

PPI-count

Selects the number of PPI available for each PE as 8(id22-27,29,30), 12(id 20-31) or 16(id 16-31)

Type

int

Default value

0x10

RAS-CFI-support

If true, fault handling interrupt for corrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-FI-support

If true, fault handling interrupt is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UE-support

If true, In-band uncorrected error reporting is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UI-support

If true, error recovery interrupt for uncorrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

SPI-blocks

Number of SPI blocks supported by the IRI, each block contains 32 SPIs

Type

int

Default value

0x1e

affinity-width

A dotted quad indicating the bitwidth of fields at each affinity level

Type

string

Default value

4.8.8.8

chip-id

Chip ID when multichip operation is enabled.

Type

int

Default value

0x0

chip-select-affinity-level

Affinity level 2 or 3 can be used for chip select

Type

int

Default value

0x3

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

direct-lpi-support

Enable support for LPI operations through GICR registers

Type

bool

Default value

0x0

enable-multichip-operation

Enables multi-chip operation between Distributors in distributed GIC IRI

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

gict-allow-ns-reset

If true, non-secure read/write access to GICT register is allowed at reset. Not allowed otherwise. This emulates gict_allow_ns tie-off signal.

Type

bool

Default value

0x1

max-cores-supported-by-GCI

GCI can support 1-64 cores. Maximum supported cores in GCI is configurable by device

Type

int

Default value

0x8

max-pe-on-chip

Maximum number of cores on any single chip. This will be used to identify the target chip and core when multichip operation is enabled.

Type

int

Default value

0x4

multichip-threaded-dgi

Enable sending multichip DGI messages in a separate thread. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x1

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

redistributor-group

Redistributor grouping information with affinity as JSON : {"0":["0.0.0.0","0.0.0.1"],"1":["0.0.1.0","0.0.1.1"]} where RD with quad 0.0.0.0 and 0.0.0.1 belongs to RD group 0. All the RDs belong to one group when this parameter is not given.

Type

string

Default value**redistributor-group-file**

File path to redistributor grouping information with affinity as JSON. The file uses the same format as "redistributor-group" parameter.

Type

string

Default value**redistributor-power-managed-by-pwrr**

GIC600 dedistributor power management is done by updating GICR_PWRR register

Type

bool

Default value

0x1

reg-base

GIC-600 base address

Type

int

Default value

0x2c010000

reg-base-per-redistributor

Base address for each redistributor in the form '0.0.0.0=0x2c010000, 0.0.0.1=0x2c020000'. All redistributors must be specified and this overrides the reg-base parameter (except that reg-base will still be used for the top-level redistributor). If reg-base-per-redistributor-file is specified, this parameter is ignored.

Type

string

Default value

3.10.33 GIC600AE_Filter

GIC-600AE IRI implementation: Single chip validation/filter component variant limited to 265 PE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-957: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC600AE_Filter

This model has the following Iris instances:

Table 3-958: GIC600AE_Filter Iris instances

InstanceName	ComponentName
GIC600AE_Filter	GIC_IRI
GIC600AE_Filter.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC600AE_Filter.ITS0	GICv3InterruptTranslationService

InstanceName	ComponentName
GIC600AE_Filter.rd_0	GICv3RedistributorInternal
GIC600AE_Filter.rd_0_0	GICv3RedistributorInternal
GIC600AE_Filter.rd_0_0_0	GICv3RedistributorInternal
GIC600AE_Filter.rd_0_0_0_0	GICv3Redistributor
GIC600AE_Filter.rd_t1	GICv3Distributor

This model has the following MTI trace components:

Table 3-959: GIC600AE_Filter MTI instances

InstanceName	ComponentName
GIC600AE_Filter	GICv3IRI
GIC600AE_Filter.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC600AE_Filter.ITS0	GICv3InterruptTranslationService
GIC600AE_Filter.rd_0	GICv3RedistributorInternal
GIC600AE_Filter.rd_0_0	GICv3RedistributorInternal
GIC600AE_Filter.rd_0_0_0	GICv3RedistributorInternal
GIC600AE_Filter.rd_0_0_0_0	GICv3Redistributor
GIC600AE_Filter.rd_t1	GICv3Distributor

GIC600AE_Filter contains the following CADI targets:

- GIC600AE

About GIC600AE_Filter

GIC600AE and GIC600AE_Filter are minimal models of an Arm® GIC-600AE Generic Interrupt Controller, suitable for single-chip or multichip systems, where multichip operation has been tested, with the following exceptions:

- LPI/LPI command is implemented but untested.
- Power operation is unimplemented.

They provide a simple configuration interface that allows designers to introduce GIC600AE-like functionality to their systems, while only implementing the architectural behavior, as defined by the GICv3 architecture.

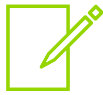
All implementation-specific registers and functionality are implemented.

As with the other GIC components, there are two variants of the model with slightly different memory interfaces. Both GIC600AE and GIC600AE_Filter have a `pvbuss_s` port for register accesses and a `pvbuss_m` port for the LPI-related traffic from redistributors and the ITS. The `pvbuss_m` port is also used to compose multichip operation.

In addition, the GIC600AE_Filter variant has a `pvbuss_filtermiss_m` port, to which any transactions on the `pvbuss_s` port and not directed to a 4K page used by the GIC are forwarded unmodified. Such transactions are terminated in the component when using the GIC600AE variant.

It is recommended to use the GIC600AE variant in most cases.

Also, the RAS register is implemented and various RAS errors are reported through status registers. Not all the RAS errors are available because Fast Models does not model errors that can cause ECC errors. Fast Models supports error record 0 and error record 13+. FMU registers are also implemented and are accessible by software.



Note

- Set the `FASTSIM_GIC_MEMORY_MAP` environment variable to 1 to print to `stderr` the memory map of any GICv3 or later models that are included in the platform being run.
- For writes to `GITS_TRANSLATE64R`, the `MSIRewriter` component can be used to create the correct data transactions based on the device ID. For more details about `GITS_TRANSLATE64R`, see [3.3.4 MSIRewriter](#) on page 139.

MasterID, ExtendedID, and UserFlags

This table shows how this model encodes information in the `MasterID`, `ExtendedID`, and `UserFlags` bus attributes:

Table 3-960: `pvbus_s` port

PVBus attribute	Bits used	Property encoded	Notes
<code>MasterID</code>	Bits[31:0]	Master ID that invoked the register access	-
<code>ExtendedID</code>	-	-	Not used.
<code>UserFlags</code>	-	-	Not used.



Note

The `pvbus_m` port does not use `MasterID`, `ExtendedID`, or `UserFlags`.

AE-specific features implemented

`GIC600AE_Filter` is a Functional Safety (FuSa) variant of `GIC600_Filter`. It has the following differences from `GIC600_Filter`:

- Both `GIC600AE_Filter` and `GIC600_Filter` support RAS, but only `GIC600AE_Filter` supports the Fault Management Unit (FMU). This support is controlled by the `has-fmu` parameter which is true by default. The FMU resides in the Distributor and processes faults that are detected by the Safety Mechanisms from all blocks.
- In `GIC600AE_Filter`, the GIC reset pin is connected to FMU reset. Only reset changes the error record registers and `FMU_ERRGSR` to their reset values.
- In `GIC600AE_Filter`, the Safety Mechanism detects faults and forwards them to the FMU. The FMU forwards all errors to the Safety Island.
- The APB port has been added to `GIC600AE_Filter` for FuSa purposes. It does not exist on the `GIC600_Filter`.

- GIC600AE_Filter supports both Error Recovery Interrupt (ERI) and Fault Handling Interrupt (FHI).
- GIC600AE_Filter has the limitation that it only supports error injection through the FMU_SMINJERR register.

Ports for GIC600AE_Filter

Table 3-961: Ports

Name	Protocol	Type	Description
apb_bus	PVBus	Slave	APB4 port to FMU, which will be connected from safety island for FuSa
chip_id	Value	Slave	chip_id port used for multichip operation
cpu_active_s[256]	Signal	Slave	CPUActive pins.
fm_u_error_int	Signal	Master	FuSa FMU error interrupt signal
fm_u_fault_int	Signal	Master	FuSa FMU fault interrupt signal
po_reset	Signal	Slave	Reset.
ppi_in_<n>[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
pvbus_filtermiss_m	PVBus	Master	Memory bus out. Transactions not filtered by the component.
pvbus_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Reset.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
wake_request[256]	Signal	Master	Power management outputs.

Parameters for GIC600AE_Filter

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value

DS-behaviour

GICD_CTLR.DS field behaviour: 0:RAZ/WI, 1:RAO/WI, 2:RW

Type

int

Default value

0x2

IIDR

GICD_IIDR and GICR_IIDR value. When the parameter is not given or is set to zero, the GIC600 model is configured to have the latest revision.

Type

int

Default value

0x0

ITS-ID-bits

Number of interrupt bits supported by ITS.

Type

int

Default value

0x10

ITS-collection-ID-bits

Number of collection bits supported by ITS (optional parameter, 0 => 16bits support and GITS_TYPER.CIL=0)

Type

int

Default value

0x8

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

PPI-count

Selects the number of PPI available for each PE as 8(id22-27,29,30), 12(id 20-31) or 16(id 16-31)

Type

int

Default value

0x10

RAS-CFI-support

If true, fault handling interrupt for corrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-FI-support

If true, fault handling interrupt is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UE-support

If true, In-band uncorrected error reporing is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UI-support

If true, error recovery interrupt for uncorrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

SPI-blocks

Number of SPI blocks supported by the IRI, each block contains 32 SPIs

Type

int

Default value

0x1e

affinity-width

A dotted quad indicating the bitwidth of fields at each affinity level

Type

string

Default value

4.8.8.8

chip-id

Chip ID when multichip operation is enabled.

Type

int

Default value

0x0

chip-select-affinity-level

Affinity level 2 or 3 can be used for chip select

Type

int

Default value

0x3

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

direct-lpi-support

Enable support for LPI operations through GICR registers

Type

bool

Default value

0x0

enable-multichip-operation

Enables multi-chip operation between Distributors in distributed GIC IRI

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

gict-allow-ns-reset

If true, non-secure read/write access to GICT register is allowed at reset. Not allowed otherwise. This emulates gict_allow_ns tie-off signal.

Type

bool

Default value

0x1

max-cores-supported-by-GCI

GCI can support 1-64 cores. Maximum supported cores in GCI is configurable by device

Type

int

Default value

0x8

max-pe-on-chip

Maximum number of cores on any single chip. This will be used to identify the target chip and core when multichip operation is enabled.

Type

int

Default value

0x4

multichip-threaded-dgi

Enable sending multichip DGI messages in a separate thread. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x1

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

redistributor-group

Redistributor grouping information with affinity as JSON : {"0":["0.0.0.0","0.0.0.1"],"1":["0.0.1.0","0.0.1.1"]} where RD with quad 0.0.0.0 and 0.0.0.1 belongs to RD group 0. All the RDs belong to one group when this parameter is not given.

Type

string

Default value**redistributor-group-file**

File path to redistributor grouping information with affinity as JSON. The file uses the same format as "redistributor-group" parameter.

Type

string

Default value**redistributor-power-managed-by-pwrr**

GIC600 dedistributor power management is done by updating GICR_PWRR register

Type

bool

Default value

0x1

reg-base

GIC-600 base address

Type

int

Default value

0x2c010000

reg-base-per-redistributor

Base address for each redistributor in the form '0.0.0.0=0x2c010000, 0.0.0.1=0x2c020000'. All redistributors must be specified and this overrides the reg-base parameter (except that reg-base will still be used for the top-level redistributor). If reg-base-per-redistributor-file is specified, this parameter is ignored.

Type

string

Default value

3.10.34 GIC600_Filter

GIC-600 IRI implementation: Single chip validation/filter component variant limited to 265 PE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-962: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC600_Filter

This model has the following Iris instances:

Table 3-963: GIC600_Filter Iris instances

InstanceName	ComponentName
GIC600_Filter	GIC_IRI
GIC600_Filter.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC600_Filter.ITS0	GICv3InterruptTranslationService
GIC600_Filter.rd_0	GICv3RedistributorInternal
GIC600_Filter.rd_0_0	GICv3RedistributorInternal
GIC600_Filter.rd_0_0_0	GICv3RedistributorInternal
GIC600_Filter.rd_0_0_0_0	GICv3Redistributor
GIC600_Filter.rd_tl	GICv3Distributor

This model has the following MTI trace components:

Table 3-964: GIC600_Filter MTI instances

InstanceName	ComponentName
GIC600_Filter	GICv3IRI
GIC600_Filter.GICV3_ProtocolChecker	GICv3ProtocolChecker

InstanceName	ComponentName
GIC600_Filter.ITS0	GICv3InterruptTranslationService
GIC600_Filter.rd_0	GICv3RedistributorInternal
GIC600_Filter.rd_0_0	GICv3RedistributorInternal
GIC600_Filter.rd_0_0_0	GICv3RedistributorInternal
GIC600_Filter.rd_0_0_0_0	GICv3Redistributor
GIC600_Filter.rd_tl	GICv3Distributor

GIC600_Filter contains the following CADI targets:

- GIC600

About GIC600_Filter

GIC600 and GIC600_Filter are minimal models of an Arm® GIC-600 Generic Interrupt Controller, suitable for single-chip or multichip systems, where multichip operation has been tested, with the following exceptions:

- LPI/LPI command is implemented but untested.
- Power operation is unimplemented.

They provide a simple configuration interface that allows designers to introduce GIC600-like functionality to their systems, while only implementing the architectural behavior, as defined by the GICv3 architecture.

All implementation-specific registers and functionality are implemented.

As with the other GIC components, there are two variants of the model with slightly different memory interfaces. Both GIC600 and GIC600_Filter have a `pvbuss_s` port for register accesses and a `pvbuss_m` port for the LPI-related traffic from redistributors and the ITS. The `pvbuss_m` port is also used to compose multichip operation.

In addition, the GIC600_Filter variant has a `pvbuss_filtermiss_m` port, to which any transactions on the `pvbuss_s` port and not directed to a 4K page used by the GIC are forwarded unmodified. Such transactions are terminated in the component when using the GIC600 variant.

It is recommended to use the GIC600 variant in most cases.

Also, the RAS register is implemented and various RAS errors are reported through status registers. Not all the RAS errors are available because Fast Models does not model errors that can cause ECC errors. Fast Models supports error record 0 and error record 13+.



Note

- Set the `FASTSIM_GIC_MEMORY_MAP` environment variable to 1 to print to `stderr` the memory map of any GICv3 or later models that are included in the platform being run.

- For writes to GITS_TRANSLATE64R, the MSIRewriter component can be used to create the correct data transactions based on the device ID. For more details about GITS_TRANSLATE64R, see [3.3.4 MSIRewriter](#) on page 139.

MasterID, ExtendedID, and UserFlags

This table shows how this model encodes information in the `MasterID`, `ExtendedID`, and `UserFlags` bus attributes:

Table 3-965: pvbus_s port

PVBus attribute	Bits used	Property encoded	Notes
MasterID	Bits[31:0]	Master ID that invoked the register access	-
ExtendedID	-	-	Not used.
UserFlags	-	-	Not used.



The `pvbus_m` port does not use `MasterID`, `ExtendedID`, or `UserFlags`.

Ports for GIC600_Filter

Table 3-966: Ports

Name	Protocol	Type	Description
chip_id	Value	Slave	chip_id port which is valid from GIC600 r1p2. Writing to this port for prior GIC600 version will be ignored.
cpu_active_s[256]	Signal	Slave	CPUActive pins.
po_reset	Signal	Slave	Reset.
ppi_in_<n>[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
pvbus_filtermiss_m	PVBus	Master	Memory bus out. Transactions not filtered by the component.
pvbus_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Reset.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
wake_request[256]	Signal	Master	Power management outputs.

Parameters for GIC600_Filter

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value**DS-behaviour**

GICD_CTLR.DS field behaviour: 0:RAZ/WI, 1:RAO/WI, 2:RW

Type

int

Default value

0x2

IIDR

GICD_IIDR and GICR_IIDR value. When the parameter is not given or is set to zero, the GIC600 model is configured to have the latest revision.

Type

int

Default value

0x0

ITS-ID-bits

Number of interrupt bits supported by ITS.

Type

int

Default value

0x10

ITS-collection-ID-bits

Number of collection bits supported by ITS (optional parameter, 0 => 16bits support and GITS_TYPER.CIL=0)

Type

int

Default value

0x8

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

PPI-count

Selects the number of PPI available for each PE as 8(id22-27,29,30), 12(id 20-31) or 16(id 16-31)

Type

int

Default value

0x10

RAS-CFI-support

If true, fault handling interrupt for corrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-FI-support

If true, fault handling interrupt is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UE-support

If true, In-band uncorrected error reporting is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UI-support

If true, error recovery interrupt for uncorrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

SPI-blocks

Number of SPI blocks supported by the IRI, each block contains 32 SPIs

Type

int

Default value

0x1e

affinity-width

A dotted quad indicating the bitwidth of fields at each affinity level

Type

string

Default value

4.8.8.8

chip-id

Chip ID when multichip operation is enabled.

Type

int

Default value

0x0

chip-select-affinity-level

Affinity level 2 or 3 can be used for chip select

Type

int

Default value

0x3

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

direct-lpi-support

Enable support for LPI operations through GICR registers

Type

bool

Default value

0x0

enable-multichip-operation

Enables multi-chip operation between Distributors in distributed GIC IRI

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

gict-allow-ns-reset

If true, non-secure read/write access to GICT register is allowed at reset. Not allowed otherwise. This emulates gict_allow_ns tie-off signal.

Type

bool

Default value

0x1

max-cores-supported-by-GCI

GCI can support 1-64 cores. Maximum supported cores in GCI is configurable by device

Type

int

Default value

0x8

max-pe-on-chip

Maximum number of cores on any single chip. This will be used to identify the target chip and core when multichip operation is enabled.

Type

int

Default value

0x4

multichip-threaded-dgi

Enable sending multichip DGI messages in a separate thread. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x1

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

redistributor-group

Redistributor grouping information with affinity as JSON : {"0":["0.0.0.0","0.0.0.1"],"1":["0.0.1.0","0.0.1.1"]} where RD with quad 0.0.0.0 and 0.0.0.1 belongs to RD group 0. All the RDs belong to one group when this parameter is not given.

Type

string

Default value**redistributor-group-file**

File path to redistributor grouping information with affinity as JSON. The file uses the same format as "redistributor-group" parameter.

Type

string

Default value**redistributor-power-managed-by-pwrr**

GIC600 dedistributor power management is done by updating GICR_PWRR register

Type

bool

Default value

0x1

reg-base

GIC-600 base address

Type

int

Default value

0x2c010000

reg-base-per-redistributor

Base address for each redistributor in the form '0.0.0.0=0x2c010000, 0.0.0.1=0x2c020000'. All redistributors must be specified and this overrides the reg-base parameter (except that reg-base will still be used for the top-level redistributor). If reg-base-per-redistributor-file is specified, this parameter is ignored.

Type

string

Default value

3.10.35 GIC625

GIC-625 IRI implementation: Single chip validation/filter component variant limited to 8 PE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-967: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC625

This model has the following Iris instances:

Table 3-968: GIC625 Iris instances

InstanceName	ComponentName
GIC625	GIC_IRI
GIC625.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC625.rd_0	GICv3RedistributorInternal

InstanceName	ComponentName
GIC625.rd_0_0	GICv3RedistributorInternal
GIC625.rd_0_0_0	GICv3RedistributorInternal
GIC625.rd_0_0_0_0	GICv3Redistributor
GIC625.rd_t1	GICv3Distributor

This model has the following MTI trace components:

Table 3-969: GIC625 MTI instances

InstanceName	ComponentName
GIC625	GICv3IRI
GIC625.GICv3_ProtocolChecker	GICv3ProtocolChecker
GIC625.rd_0	GICv3RedistributorInternal
GIC625.rd_0_0	GICv3RedistributorInternal
GIC625.rd_0_0_0	GICv3RedistributorInternal
GIC625.rd_0_0_0_0	GICv3Redistributor
GIC625.rd_t1	GICv3Distributor

GIC625 contains the following CADI targets:

- GIC625

About GIC625

GIC625 and GIC625_Filter are Generic Interrupt Controllers that handle interrupts from peripherals to cores and interrupts between cores in a single cluster of up to 8 cores. They support the GICv3 and GICv3.1 architectures.

GIC625 and GIC625_Filter support the following features:

- Grouping cores in GCI.
- LPI is disabled.
- Power Management updates such as:
 - GCI grouping.
 - GCI RDPowerDown support
 - Updating the GCI ID for the cluster.
 - GCI GICR_PWRR register updates.
- Inversion of SPI and PPI inputs.
- Extended PPI.



Note

- Extended SPI is not supported.
- All implementation-specific registers and functionality are implemented.

Ports for GIC625

Table 3-970: Ports

Name	Protocol	Type	Description
chip_id	Value	Slave	chip_id port used for multichip operation
cpu_active_s[8]	Signal	Slave	CPUActive pins.
po_reset	Signal	Slave	Resets. This is used as a dbg_reset in TRM, and also be used for cold reset for PMU and FMU.
ppi_in_<n>[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
pvbus_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[8]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Resets.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
wake_request[8]	Signal	Master	Power management outputs.

Parameters for GIC625

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value

DS-behaviour

GICD_CTLR.DS field behaviour: 0:RAZ/WI, 1:RAO/WI, 2:RW

Type

int

Default value

0x2

IIDR

GICD_IIDR and GICR_IIDR value. When the parameter is not given or is set to zero, the GIC600 model is configured to have the latest revision.

Type

int

Default value

0x0

PPI-count

Selects the number of PPI available for each PE as 8(id22-27,29,30), 12(id 20-31) or 16(id 16-31)

Type

int

Default value

0x10

RAS-CFI-support

If true, fault handling interrupt for corrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-FI-support

If true, fault handling interrupt is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UE-support

If true, In-band uncorrected error reporting is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UI-support

If true, error recovery interrupt for uncorrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

SPI-blocks

Number of SPI blocks supported by the IRI, each block contains 32 SPIs

Type

int

Default value

0x1e

affinity-width

A dotted quad indicating the bitwidth of fields at each affinity level

Type

string

Default value

4.8.8.8

chip-select-affinity-level

Affinity level 2 or 3 can be used for chip select

Type

int

Default value

0x3

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

gict-allow-ns-reset

If true, non-secure read/write access to GICT register is allowed at reset. Not allowed otherwise. This emulates gict_allow_ns tie-off signal.

Type

bool

Default value

0x1

max-cores-supported-by-GCI

GCI can support 1-64 cores. Maximum supported cores in GCI is configurable by device

Type

int

Default value

0x8

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

redistributor-group

Redistributor grouping information with affinity as JSON : {"0":["0.0.0.0","0.0.0.1"],"1":["0.0.1.0","0.0.1.1"]} where RD with quad 0.0.0.0 and 0.0.0.1 belongs to RD group 0. All the RDs belong to one group when this parameter is not given.

Type

string

Default value**redistributor-group-file**

File path to redistributor grouping information with affinity as JSON. The file uses the same format as "redistributor-group" parameter.

Type

string

Default value**redistributor-power-managed-by-pwrr**

GIC600 dedistributor power management is done by updating GICR_PWRR register

Type

bool

Default value

0x1

reg-base

GIC-600 base address

Type

int

Default value

0x2c010000

reg-base-per-redistributor

Base address for each redistributor in the form '0.0.0.0=0x2c010000, 0.0.0.1=0x2c020000'. All redistributors must be specified and this overrides the reg-base parameter (except that reg-base will still be used for the top-level redistributor). If reg-base-per-redistributor-file is specified, this parameter is ignored.

Type

string

Default value

3.10.36 GIC625_Filter

GIC-625 IRI implementation: Single chip validation/filter component variant limited to 8 PE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-971: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC625_Filter

This model has the following Iris instances:

Table 3-972: GIC625_Filter Iris instances

InstanceName	ComponentName
GIC625_Filter	GIC_IRI
GIC625_Filter.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC625_Filter.rd_0	GICv3RedistributorInternal
GIC625_Filter.rd_0_0	GICv3RedistributorInternal
GIC625_Filter.rd_0_0_0	GICv3RedistributorInternal

InstanceName	ComponentName
GIC625_Filter.rd_0_0_0_0	GICv3Redistributor
GIC625_Filter.rd_tl	GICv3Distributor

This model has the following MTI trace components:

Table 3-973: GIC625_Filter MTI instances

InstanceName	ComponentName
GIC625_Filter	GICv3IRI
GIC625_Filter.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC625_Filter.rd_0	GICv3RedistributorInternal
GIC625_Filter.rd_0_0	GICv3RedistributorInternal
GIC625_Filter.rd_0_0_0	GICv3RedistributorInternal
GIC625_Filter.rd_0_0_0_0	GICv3Redistributor
GIC625_Filter.rd_tl	GICv3Distributor

GIC625_Filter contains the following CADI targets:

- GIC625

About GIC625_Filter

GIC625 and GIC625_Filter are Generic Interrupt Controllers that handle interrupts from peripherals to cores and interrupts between cores in a single cluster of up to 8 cores. They support the GICv3 and GICv3.1 architectures.

GIC625 and GIC625_Filter support the following features:

- Grouping cores in GCI.
- LPI is disabled.
- Power Management updates such as:
 - GCI grouping.
 - GCI RDPowerDown support
 - Updating the GCI ID for the cluster.
 - GCI GICR_PWRR register updates.
- Inversion of SPI and PPI inputs.
- Extended PPI.



- Extended SPI is not supported.
- All implementation-specific registers and functionality are implemented.

Ports for GIC625_Filter

Table 3-974: Ports

Name	Protocol	Type	Description
chip_id	Value	Slave	chip_id port used for multichip operation
cpu_active_s[8]	Signal	Slave	CPUActive pins.
po_reset	Signal	Slave	Reset.
ppi_in_<n>[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
pvbus_filtermiss_m	PVBus	Master	Memory bus out. Transactions not filtered by the component.
pvbus_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[8]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Reset.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
wake_request[8]	Signal	Master	Power management outputs.

Parameters for GIC625_Filter

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value

DS-behaviour

GICD_CTLR.DS field behaviour: 0:RAZ/WI, 1:RAO/WI, 2:RW

Type

int

Default value

0x2

IIDR

GICD_IIDR and GICR_IIDR value. When the parameter is not given or is set to zero, the GIC600 model is configured to have the latest revision.

Type

int

Default value

0x0

PPI-count

Selects the number of PPI available for each PE as 8(id22-27,29,30), 12(id 20-31) or 16(id 16-31)

Type

int

Default value

0x10

RAS-CFI-support

If true, fault handling interrupt for corrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-FI-support

If true, fault handling interrupt is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UE-support

If true, In-band uncorrected error reporting is supported. Not supported otherwise.

Type

bool

Default value

0x0

RAS-UI-support

If true, error recovery interrupt for uncorrected errors is supported. Not supported otherwise.

Type

bool

Default value

0x0

SPI-blocks

Number of SPI blocks supported by the IRI, each block contains 32 SPIs

Type

int

Default value

0x1e

affinity-width

A dotted quad indicating the bitwidth of fields at each affinity level

Type

string

Default value

4.8.8.8

chip-select-affinity-level

Affinity level 2 or 3 can be used for chip select

Type

int

Default value

0x3

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

gict-allow-ns-reset

If true, non-secure read/write access to GICT register is allowed at reset. Not allowed otherwise. This emulates gict_allow_ns tie-off signal.

Type

bool

Default value

0x1

max-cores-supported-by-GCI

GCI can support 1-64 cores. Maximum supported cores in GCI is configurable by device

Type

int

Default value

0x8

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

redistributor-group

Redistributor grouping information with affinity as JSON : {"0":["0.0.0.0","0.0.0.1"],"1":["0.0.1.0","0.0.1.1"]} where RD with quad 0.0.0.0 and 0.0.0.1 belongs to RD group 0. All the RDs belong to one group when this parameter is not given.

Type

string

Default value**redistributor-group-file**

File path to redistributor grouping information with affinity as JSON. The file uses the same format as "redistributor-group" parameter.

Type

string

Default value**redistributor-power-managed-by-pwrr**

GIC600 redistributor power management is done by updating GICR_PWRR register

Type

bool

Default value

0x1

reg-base

GIC-600 base address

Type

int

Default value

0x2c010000

reg-base-per-redistributor

Base address for each redistributor in the form '0.0.0.0=0x2c010000, 0.0.0.1=0x2c020000'. All redistributors must be specified and this overrides the reg-base parameter (except that reg-base will still be used for the top-level redistributor). If reg-base-per-redistributor-file is specified, this parameter is ignored.

Type

string

Default value

3.10.37 GIC700

GIC-700 IRI implementation: Single/multi chip validation/filter component variant limited to 64 PE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-975: IP revisions support

Revision	Quality level
r0p0	Full support
r1p0	Full support
r2p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC700

This model has the following Iris instances:

Table 3-976: GIC700 Iris instances

InstanceName	ComponentName
GIC700	GIC_IRI
GIC700.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC700.ITS0	GICv4InterruptTranslationService

InstanceName	ComponentName
GIC700.ITS0.bus_subordinate	PVBusSlave
GIC700.rd_0	GICv4RedistributorInternal
GIC700.rd_0_0	GICv4RedistributorInternal
GIC700.rd_0_0_0	GICv4RedistributorInternal
GIC700.rd_0_0_0_0	GICv3Redistributor
GIC700.rd_t1	GICv3Distributor

This model has the following MTI trace components:

Table 3-977: GIC700 MTI instances

InstanceName	ComponentName
GIC700	GICv3IRI
GIC700.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC700.ITS0	GICv4InterruptTranslationService
GIC700.ITS0.bus_subordinate	PVBusSlave
GIC700.rd_0	GICv4RedistributorInternal
GIC700.rd_0_0	GICv4RedistributorInternal
GIC700.rd_0_0_0	GICv4RedistributorInternal
GIC700.rd_0_0_0_0	GICv3Redistributor
GIC700.rd_t1	GICv3Distributor

GIC700 contains the following CADI targets:

- GIC700

About GIC700

GIC700 and GIC700_Filter are minimal models of an Arm® GIC-700 Generic Interrupt Controller, suitable for single-chip or multichip systems. They support the following multichip operations:

- Multichip configuration through distributor registers.
- SPI.
- SGI.
- Physical LPI.
- Physical LPI command.
- Virtual LPI.
- Virtual LPI command.

The models provide a simple configuration interface that allows designers to introduce GIC700-like functionality to systems, while only implementing the architectural behavior, as defined by the GICv3/GICv4 architecture.

All implementation-specific registers and functionality are implemented.

As with the other GIC components, there are two variants of the model with slightly different memory interfaces. Both GIC700 and GIC700_Filter have a `pvbuss_s` port for register accesses and a `pvbuss_m` port for the LPI-related traffic from redistributors and the ITS.

In addition, the GIC700_Filter variant has a `pvbuss_filtermiss_m` port, to which any transactions on the `pvbuss_s` port and not directed to a 4K page used by the GIC are forwarded unmodified. Such transactions are terminated in the component when using the GIC700 variant.

GIC700 and GIC700_Filter support an extended interrupt range, with 64 more PPIs and 1024 more SPIs. To program additional interrupts, refer to the GICv3/GICv4 specification.

GIC700 and GIC700_Filter support direct injection of virtual interrupts. This reduces the overhead of trapping to hypervisor for the generation of virtual interrupts. In the GICv3/GICv4 specification, direct injection is supported in two different versions, GICv4.0 and GICv4.1. GIC700 and GIC700_Filter support GICv4.1.

It is recommended to use GIC700 instead of GIC700_Filter in most cases.

The GIC700 model supports multiple revisions of the IP. By default, its behavior is set to the latest revision supported. Parameters can be used to enable or disable features that are specific to earlier revisions. The LCA (Local Cross-chip Addressing) feature which is added in r2p0 is disabled by default and is enabled by setting the parameter `enable-local-cross-chip-addressing` to true. The model's registers follow the specification for the latest IP revision supported.



- Set the `FASTSIM_GIC_MEMORY_MAP` environment variable to 1 to print to `stderr` the memory map of any GICv3 or later models that are included in the platform being run.
- For writes to `GITS_TRANSLATE64R`, the MSIRewriter component can be used to create the correct data transactions based on the device ID. For more details about `GITS_TRANSLATE64R`, see [3.3.4 MSIRewriter](#) on page 139.

Limitations

- The following multichip operations are not yet supported:
 - vPE control.
 - Power control.
- There is no support for the `vMOV` command.
- Limited testing has been performed for multichip LPI operations, so the model might contain defects.
- One ITS is supported for each GIC for multichip operations.

MasterID, ExtendedID, and UserFlags

This table shows how this model encodes information in the `MasterID`, `ExtendedID`, and `UserFlags` bus attributes:

Table 3-978: pvbus_s port

PVBus attribute	Bits used	Property encoded	Notes
MasterID	Bits[31:0]	Master ID that invoked the register access	-
ExtendedID	-	-	Not used.
UserFlags	-	-	Not used.



The pvbus_m port does not use MasterID, ExtendedID, Or UserFlags.

Ports for GIC700

Table 3-979: Ports

Name	Protocol	Type	Description
axi_stream_msi_s[32]	PVBus	Slave	AXI Stream Interface ports for MSI (message-signalled interrupt) translation, in GIC-700. Typically the SMMU's TCU connects to this port for MSI.
chip_id	Value	Slave	chip_id port for multichip operation
cpu_active_s[256]	Signal	Slave	CPUActive pins.
cpu_wake_request[256]	Signal	Master	-
extended_ppi_in_<n>[64]	Signal	Slave	Extended private peripheral interrupts (ID1056-ID1119) for cpu <n>, where <n> is in the range 0-255.
extended_spi_in[1024]	Signal	Slave	Extended Shared peripheral interrupts.
icdrt_out	PVBus	Master	AXI Stream Interface. This is used only when local cross chip addressing is enabled at the moment
icrdt_in	PVBus	Slave	-
po_reset	Signal	Slave	Resets.
ppi_in_<n>[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
pvbus_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Resets.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
wake_request[256]	Signal	Master	Power management outputs.

Parameters for GIC700

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value**DS-behaviour**

GICD_CTLR.DS field behaviour: 0:RAZ/WI, 1:RAO/WI, 2:RW

Type

int

Default value

0x2

GICD_CTLR-DS-1-means-secure-only

If GICD_CTLR.DS=1, GICD supports a single security state which is secure if this is true, otherwise is non-secure.

Type

bool

Default value

0x0

GICD_TYPER2

GICD_TYPER2 value containing VID and VIL to define the width of vPEID for GICv4.1.

Type

int

Default value

0x0

IIDR

GICD_IIDR and GICR_IIDR value

Type

int

Default value

0x0

ITS-ID-bits

Number of interrupt bits supported by ITS.

Type

int

Default value

0x10

ITS-collection-ID-bits

Number of collection bits supported by ITS (optional parameter, 0 => 16bits support and GITS_TYPER.CIL=0)

Type

int

Default value

0x8

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

ITS-enable-itt-address-verification

If true, a transaction will be sent to ITT Address for verification.

Type

bool

Default value

0x0

ITS-hardware-collection-count

Number of hardware collections held exclusively in the ITS

Type

int

Default value

0x0

ITS-shared-vPE-table

Number of affinity levels to which the vPE configuration table is shared. This parameter is valid when has-gicv4.1 is true.

Type

int

Default value

0x0

ITS-vmovp-bit

Device supports software issuing a VMOVP to only one of the ITSs that has a mapping for a vPE. The device itself ensures synchronization of the VMOVP command across all ITSs that have mapping for that vPE.

Type

bool

Default value

0x0

PPI-count

Selects the number of PPI available for each PE as 8(id22-27,29,30), 12(id 20-31) or 16(id 16-31)

Type

int

Default value

0x10

SPI-blocks

Number of SPI blocks supported by the IRI, each block contains 32 SPIs

Type

int

Default value

0x1e

add-output-cpu-wake-request-signal-from-redistributor

if true, the redistributor will have the output signal cpu_wake_request from GIC to DSU and if false, the signals are not added to the redistributor

Type

bool

Default value

0x0

affinity-width

A dotted quad indicating the bitwidth of fields at each affinity level

Type

string

Default value

4.8.8.8

allow-LPIEN-clear

Allow RW behaviour on GICR_CTLR.LPIEN instead of set once

Type

bool

Default value

0x1

chip-count

The total number of chips supported.

Type

int

Default value

0x10

chip-id

Chip ID when multichip operation is enabled.

Type

int

Default value

0x0

chip-select-affinity-level

Affinity level 2 or 3 can be used for chip select

Type

int

Default value

0x3

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

common-vPE-table-affinity

Affinity value list in the form of 'a.b.c.d, e.f.g.h, etc' given to the ITS(s) (where a.b.c.d corresponds to ITS0, e.f.g.h corresponds to ITS1 and so on). Under an affinity value the vPE configuration table is shared among redistributors where the level to be shared is defined by ITS-shared-vPE-table. This parameter is valid when has-gicv4.1 is true.

Type

string

Default value**consolidators**

Specify consolidators' base addresses, interrupt line counts and base interrupt IDs, in the form 'baseAddr0:itlineCount0:baseINTID0, baseAddr1:itlineCount1:baseINTID1, [etc]' (eg '0x3f100000:64:4096, 0x3f200000:64:4224'). The consolidators' count is inferred from the list (maximum of 4). If not specified, the component contains no consolidators.

Type

string

Default value**cross-chip-AMBA-is-ACE**

Indicates the AMBA protocol that the cross-chip interface uses. If true, the cross-chip interface uses the ACE5-Lite protocol. Otherwise, it uses the AXI5-Stream protocol. This parameter is meaningful only if enable-multichip-operation is set.

Type

bool

Default value

0x0

direct-lpi-support

Enable support for LPI operations through GICR registers

Type

bool

Default value

0x0

enable-local-cross-chip-addressing

If true, each distributor in a multichip system will use routing table address (ADDR) values programmed by local software, and will not be overwritten by the default chip. Otherwise, all the distributors will share the same routing table address values programmed in the default chip. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

extended-ppi-count

Number of extended PPI supported

Type

int

Default value

0x40

extended-spi-count

Number of extended SPI supported

Type

int

Default value

0x400

has-gicv4.1

Enable GICv4.1 functionality; when false the component is inactive.

Type

bool

Default value

0x1

has-two-security-states

If true, has two security states

Type

bool

Default value

0x1

max-cores-supported-by-GCI

GCI can support 1-64 cores. Maximum supported cores in GCI is configurable by device

Type

int

Default value

0x8

max-pe-on-chip

Maximum number of cores on any single chip. This will be used to identify the target chip and core.

Type

int

Default value

0x4

multichip-threaded-dgi

Enable sending multichip DGI messages in a separate thread. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x1

output_attributes

User-defined transform to be applied to bus attributes like MasterID, ExtendedID or UserFlags. Currently, only works for MPAM Attributes encoding into bus attributes.

Type

string

Default value

ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID,
ExtendedID[38]=MPAM_NS

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

prog-mpidr

Whether software or hardware can remove cores from a GIC configuration. (0: none, 1: prog - Secure software to remove cores during the boot up of a system. 2: strap - enables hardware to remove cores as GIC exits reset)

Type

int

Default value

0x0

redistributor-group

Redistributor grouping information with affinity as JSON : {"0":["0.0.0.0","0.0.0.1"],"1":["0.0.1.0","0.0.1.1"]} where RD with quad 0.0.0.0 and 0.0.0.1 belongs to RD group 0. All the RDs belong to one group when this parameter is not given.

Type

string

Default value**redistributor-group-file**

File path to redistributor grouping information with affinity as JSON. The file uses the same format as "redistributor-group" parameter.

Type

string

Default value**redistributor-power-managed-by-pwrr**

GIC-700 dedistributor power management is done by updating GICR_PWRR register

Type

bool

Default value

0x1

reg-base

GIC-700 base address

Type

int

Default value

0x2c010000

3.10.38 GIC700_Filter

GIC-700 IRI implementation: Single/multi chip validation/filter component variant limited to 64 PE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-980: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC700_Filter

This model has the following Iris instances:

Table 3-981: GIC700_Filter Iris instances

InstanceName	ComponentName
GIC700_Filter	GIC_IRI
GIC700_Filter.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC700_Filter.ITS0	GICv4InterruptTranslationService
GIC700_Filter.ITS0.bus_subordinate	PVBusSlave
GIC700_Filter.rd_0	GICv4RedistributorInternal
GIC700_Filter.rd_0_0	GICv4RedistributorInternal
GIC700_Filter.rd_0_0_0	GICv4RedistributorInternal
GIC700_Filter.rd_0_0_0_0	GICv3Redistributor
GIC700_Filter.rd_t1	GICv3Distributor

This model has the following MTI trace components:

Table 3-982: GIC700_Filter MTI instances

InstanceName	ComponentName
GIC700_Filter	GICv3IRI
GIC700_Filter.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC700_Filter.ITS0	GICv4InterruptTranslationService
GIC700_Filter.ITS0.bus_subordinate	PVBusSlave
GIC700_Filter.rd_0	GICv4RedistributorInternal
GIC700_Filter.rd_0_0	GICv4RedistributorInternal
GIC700_Filter.rd_0_0_0	GICv4RedistributorInternal
GIC700_Filter.rd_0_0_0_0	GICv3Redistributor
GIC700_Filter.rd_t1	GICv3Distributor

GIC700_Filter contains the following CADI targets:

- GIC700

About GIC700_Filter

GIC700 and GIC700_Filter are minimal models of an Arm® GIC-700 Generic Interrupt Controller, suitable for single-chip or multichip systems. They support the following multichip operations:

- Multichip configuration through distributor registers.

- SPI.
- SGI.
- Physical LPI.
- Physical LPI command.
- Virtual LPI.
- Virtual LPI command.

The models provide a simple configuration interface that allows designers to introduce GIC700-like functionality to systems, while only implementing the architectural behavior, as defined by the GICv3/GICv4 architecture.

All implementation-specific registers and functionality are implemented.

As with the other GIC components, there are two variants of the model with slightly different memory interfaces. Both GIC700 and GIC700_Filter have a `pvbuss_s` port for register accesses and a `pvbuss_m` port for the LPI-related traffic from redistributors and the ITS.

In addition, the GIC700_Filter variant has a `pvbuss_filtermiss_m` port, to which any transactions on the `pvbuss_s` port and not directed to a 4K page used by the GIC are forwarded unmodified. Such transactions are terminated in the component when using the GIC700 variant.

GIC700 and GIC700_Filter support an extended interrupt range, with 64 more PPIs and 1024 more SPIs. To program additional interrupts, refer to the GICv3/GICv4 specification.

GIC700 and GIC700_Filter support direct injection of virtual interrupts. This reduces the overhead of trapping to hypervisor for the generation of virtual interrupts. In the GICv3/GICv4 specification, direct injection is supported in two different versions, GICv4.0 and GICv4.1. GIC700 and GIC700_Filter support GICv4.1.

It is recommended to use GIC700 instead of GIC700_Filter in most cases.

The GIC700 model supports multiple revisions of the IP. By default, its behavior is set to the latest revision supported. Parameters can be used to enable or disable features that are specific to earlier revisions. The LCA (Local Cross-chip Addressing) feature which is added in r2p0 is disabled by default and is enabled by setting the parameter `enable-local-cross-chip-addressing` to true. The model's registers follow the specification for the latest IP revision supported.



- Set the `FASTSIM_GIC_MEMORY_MAP` environment variable to 1 to print to `stderr` the memory map of any GICv3 or later models that are included in the platform being run.
 - For writes to `GITS_TRANSLATE64R`, the MSIRewriter component can be used to create the correct data transactions based on the device ID. For more details about `GITS_TRANSLATE64R`, see [3.3.4 MSIRewriter](#) on page 139.
-

Limitations

- The following multichip operations are not yet supported:
 - vPE control.
 - Power control.
- There is no support for the `vmov` command.
- Limited testing has been performed for multichip LPI operations, so the model might contain defects.
- One ITS is supported for each GIC for multichip operations.

MasterID, ExtendedID, and UserFlags

This table shows how this model encodes information in the `MasterID`, `ExtendedID`, and `UserFlags` bus attributes:

Table 3-983: `pvbus_s` port

PVBus attribute	Bits used	Property encoded	Notes
<code>MasterID</code>	Bits[31:0]	Master ID that invoked the register access	-
<code>ExtendedID</code>	-	-	Not used.
<code>UserFlags</code>	-	-	Not used.



The `pvbus_m` port does not use `MasterID`, `ExtendedID`, or `UserFlags`.

Ports for GIC700_Filter

Table 3-984: Ports

Name	Protocol	Type	Description
<code>axi_stream_msi_s[32]</code>	PVBus	Slave	AXI Stream Interface ports for MSI (message-signalled interrupt) translation, in GIC-700. Typically the SMMU's TCU connects to this port for MSI.
<code>chip_id</code>	Value	Slave	<code>chip_id</code> port for multichip operation
<code>cpu_active_s[256]</code>	Signal	Slave	CPUActive pins.
<code>extended_ppi_in_<n>[64]</code>	Signal	Slave	Extended private peripheral interrupts (ID1056-ID1119) for cpu <n>, where <n> is in the range 0-255.
<code>extended_spi_in[1024]</code>	Signal	Slave	Extended Shared peripheral interrupts.
<code>icdrt_out</code>	PVBus	Master	AXI Stream Interface. This is used only when local cross chip addressing is enabled at the moment
<code>icrdt_in</code>	PVBus	Slave	-
<code>po_reset</code>	Signal	Slave	Reset.
<code>ppi_in_<n>[16]</code>	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
<code>pvbus_filtermiss_m</code>	PVBus	Master	Memory bus out. Transactions not filtered by the component.
<code>pvbus_m</code>	PVBus	Master	Memory bus out: transactions generated by the IRI.

Name	Protocol	Type	Description
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Reset.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
wake_request[256]	Signal	Master	Power management outputs.

Parameters for GIC700_Filter

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value

DS-behaviour

GICD_CTLR.DS field behaviour: 0:RAZ/WI, 1:RAO/WI, 2:RW

Type

int

Default value

0x2

GICD_CTLR-DS-1-means-secure-only

If GICD_CTLR.DS=1, GICD supports a single security state which is secure if this is true, otherwise is non-secure.

Type

bool

Default value

0x0

GICD_TYPER2

GICD_TYPER2 value containing VID and VIL to define the width of vPEID for GICv4.1.

Type

int

Default value

0x0

IIDR

GICD_IIDR and GICR_IIDR value

Type

int

Default value

0x0

ITS-ID-bits

Number of interrupt bits supported by ITS.

Type

int

Default value

0x10

ITS-collection-ID-bits

Number of collection bits supported by ITS (optional parameter, 0 => 16bits support and GITS_TYPER.CIL=0)

Type

int

Default value

0x8

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

ITS-enable-itt-address-verification

If true, a transaction will be sent to ITT Address for verification.

Type

bool

Default value

0x0

ITS-hardware-collection-count

Number of hardware collections held exclusively in the ITS

Type

int

Default value

0x0

ITS-shared-vPE-table

Number of affinity levels to which the vPE configuration table is shared. This parameter is valid when has-gicv4.1 is true.

Type

int

Default value

0x0

ITS-vmovp-bit

Device supports software issuing a VMOVP to only one of the ITSs that has a mapping for a vPE. The device itself ensures synchronization of the VMOVP command across all ITSs that have mapping for that vPE.

Type

bool

Default value

0x0

PPI-count

Selects the number of PPI available for each PE as 8(id22-27,29,30), 12(id 20-31) or 16(id 16-31)

Type

int

Default value

0x10

SPI-blocks

Number of SPI blocks supported by the IRI, each block contains 32 SPIs

Type

int

Default value

0x1e

add-output-cpu-wake-request-signal-from-redistributor

if true, the redistributor will have the output signal cpu_wake_request from GIC to DSU and if false, the signals are not added to the redistributor

Type

bool

Default value

0x0

affinity-width

A dotted quad indicating the bitwidth of fields at each affinity level

Type

string

Default value

4.8.8.8

allow-LPIEN-clear

Allow RW behaviour on GICR_CTLR.LPIEN instead of set once

Type

bool

Default value

0x1

chip-count

The total number of chips supported.

Type

int

Default value

0x10

chip-id

Chip ID when multichip operation is enabled.

Type

int

Default value

0x0

chip-select-affinity-level

Affinity level 2 or 3 can be used for chip select

Type

int

Default value

0x3

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

common-vPE-table-affinity

Affinity value list in the form of 'a.b.c.d, e.f.g.h, etc' given to the ITS(s) (where a.b.c.d corresponds to ITS0, e.f.g.h corresponds to ITS1 and so on). Under an affinity value the vPE configuration table is shared among redistributors where the level to be shared is defined by ITS-shared-vPE-table. This parameter is valid when has-gicv4.1 is true.

Type

string

Default value**consolidators**

Specify consolidators' base addresses, interrupt line counts and base interrupt IDs, in the form 'baseAddr0:itlineCount0:baseINTID0, baseAddr1:itlineCount1:baseINTID1, [etc]' (eg '0x3f100000:64:4096, 0x3f200000:64:4224'). The consolidators' count is inferred from the list (maximum of 4). If not specified, the component contains no consolidators.

Type

string

Default value**cross-chip-AMBA-is-ACE**

Indicates the AMBA protocol that the cross-chip interface uses. If true, the cross-chip interface uses the ACE5-Lite protocol. Otherwise, it uses the AXI5-Stream protocol. This parameter is meaningful only if enable-multichip-operation is set.

Type

bool

Default value

0x0

direct-lpi-support

Enable support for LPI operations through GICR registers

Type

bool

Default value

0x0

enable-local-cross-chip-addressing

If true, each distributor in a multichip system will use routing table address (ADDR) values programmed by local software, and will not be overwritten by the default chip. Otherwise, all the distributors will share the same routing table address values programmed in the default chip. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

extended-ppi-count

Number of extended PPI supported

Type

int

Default value

0x40

extended-spi-count

Number of extended SPI supported

Type

int

Default value

0x400

has-gicv4.1

Enable GICv4.1 functionality; when false the component is inactive.

Type

bool

Default value

0x1

has-two-security-states

If true, has two security states

Type

bool

Default value

0x1

max-cores-supported-by-GCI

GCI can support 1-64 cores. Maximum supported cores in GCI is configurable by device

Type

int

Default value

0x8

max-pe-on-chip

Maximum number of cores on any single chip. This will be used to identify the target chip and core.

Type

int

Default value

0x4

multichip-threaded-dgi

Enable sending multichip DGI messages in a separate thread. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x1

output_attributes

User-defined transform to be applied to bus attributes like MasterID, ExtendedID or UserFlags. Currently, only works for MPAM Attributes encoding into bus attributes.

Type

string

Default value

ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID,
ExtendedID[38]=MPAM_NS

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

prog-mpidr

Whether software or hardware can remove cores from a GIC configuration. (0: none, 1: prog - Secure software to remove cores during the boot up of a system. 2: strap - enables hardware to remove cores as GIC exits reset)

Type

int

Default value

0x0

redistributor-group

Redistributor grouping information with affinity as JSON : {"0":["0.0.0.0","0.0.0.1"],"1":["0.0.1.0","0.0.1.1"]} where RD with quad 0.0.0.0 and 0.0.0.1 belongs to RD group 0. All the RDs belong to one group when this parameter is not given.

Type

string

Default value**redistributor-group-file**

File path to redistributor grouping information with affinity as JSON. The file uses the same format as "redistributor-group" parameter.

Type

string

Default value**redistributor-power-managed-by-pwrr**

GIC-700 dedistributor power management is done by updating GICR_PWRR register

Type

bool

Default value

0x1

reg-base

GIC-700 base address

Type

int

Default value

0x2c010000

3.10.39 GIC720AE

GIC-720AE IRI implementation: Single/multi chip validation/filter component variant limited to 64 PE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-985: IP revisions support

Revision	Quality level
N/A	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC720AE

This model has the following Iris instances:

Table 3-986: GIC720AE Iris instances

InstanceName	ComponentName
GIC720AE	GIC_IRI
GIC720AE.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC720AE.ITS0	GICv4InterruptTranslationService
GIC720AE.ITS0.bus_subordinate	PVBusSlave
GIC720AE.fmu	ExportTest.GIC720AEfmu
GIC720AE.fmu.pvbus_slave	PVBusSlave
GIC720AE.rd_0	GICv4RedistributorInternal
GIC720AE.rd_0_0	GICv4RedistributorInternal
GIC720AE.rd_0_0_0	GICv4RedistributorInternal
GIC720AE.rd_0_0_0_0	GICv3Redistributor
GIC720AE.rd_t1	GICv3Distributor

This model has the following MTI trace components:

Table 3-987: GIC720AE MTI instances

InstanceName	ComponentName
GIC720AE	GICv3IRI

InstanceName	ComponentName
GIC720AE.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC720AE.ITS0	GICv4InterruptTranslationService
GIC720AE.ITS0.bus_subordinate	PVBusSlave
GIC720AE.fmu	ExportTest.GIC720AEfmu
GIC720AE.fmu.pvbus_slave	PVBusSlave
GIC720AE.rd_0	GICv4RedistributorInternal
GIC720AE.rd_0_0	GICv4RedistributorInternal
GIC720AE.rd_0_0_0	GICv4RedistributorInternal
GIC720AE.rd_0_0_0_0	GICv3Redistributor
GIC720AE.rd_tl	GICv3Distributor

GIC720AE contains the following CADI targets:

- GIC720AE

About GIC720AE

The model has the same functionality as GIC700, but in addition supports the following AE-specific features:

- GIC FMU
- Multiple views

It has the same limitations as [3.10.37 GIC700](#) on page 3397.

Ports for GIC720AE

Table 3-988: Ports

Name	Protocol	Type	Description
apb_bus	PVBus	Slave	FMU signals
axi_stream_msi_s[32]	PVBus	Slave	AXI Stream Interface ports for MSI (message-signalled interrupt) translation, in GIC-720AE. Typically the SMMU's TCU connects to this port for MSI.
chip_id	Value	Slave	chip_id port for multichip operation
cpu_active_s[256]	Signal	Slave	CPUActive pins.
cpu_wake_request[256]	Signal	Master	-
extended_ppi_in_<n>[64]	Signal	Slave	Extended private peripheral interrupts (ID1056-ID1119) for cpu <n>, where <n> is in the range 0-255.
extended_spi_in[1024]	Signal	Slave	Extended Shared peripheral interrupts.
fmu_cri	Signal	Master	Critical Interrupt
fmu_eri	Signal	Master	Error recovery Interrupt
icdrt_out	PVBus	Master	AXI Stream Interface. This is used only when local cross chip addressing is enabled at the moment
icrdrt_in	PVBus	Slave	-

Name	Protocol	Type	Description
po_reset	Signal	Slave	Resets.
ppi_in_<n>[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
pvbus_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Resets.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
wake_request[256]	Signal	Master	Power management outputs.

Parameters for GIC720AE

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value

DS-behaviour

GICD_CTLR.DS field behaviour: 0:RAZ/WI, 1:RAO/WI, 2:RW

Type

int

Default value

0x2

GICD_CTLR-DS-1-means-secure-only

If GICD_CTLR.DS=1, GICD supports a single security state which is secure if this is true, otherwise is non-secure.

Type

bool

Default value

0x0

GICD_TYPER2

GICD_TYPER2 value containing VID and VIL to define the width of vPEID for GICv4.1.

Type

int

Default value

0x0

IIDR

GICD_IIDR and GICR_IIDR value

Type

int

Default value

0x0

ITS-ID-bits

Number of interrupt bits supported by ITS.

Type

int

Default value

0x10

ITS-collection-ID-bits

Number of collection bits supported by ITS (optional parameter, 0 => 16bits support and GITS_TYPER.CIL=0)

Type

int

Default value

0x8

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

ITS-enable-itt-address-verification

If true, a transaction will be sent to ITT Address for verification.

Type

bool

Default value

0x0

ITS-hardware-collection-count

Number of hardware collections held exclusively in the ITS

Type

int

Default value

0x0

ITS-shared-vPE-table

Number of affinity levels to which the vPE configuration table is shared. This parameter is valid when has-gicv4.1 is true.

Type

int

Default value

0x0

ITS-vmovp-bit

Device supports software issuing a VMOVP to only one of the ITSs that has a mapping for a vPE. The device itself ensures synchronization of the VMOVP command across all ITSs that have mapping for that vPE.

Type

bool

Default value

0x0

PPI-count

Selects the number of PPI available for each PE as 8(id22-27,29,30), 12(id 20-31) or 16(id 16-31)

Type

int

Default value

0x10

SPI-blocks

Number of SPI blocks supported by the IRI, each block contains 32 SPIs

Type

int

Default value

0x1e

add-output-cpu-wake-request-signal-from-redistributor

if true, the redistributor will have the output signal `cpu_wake_request` from GIC to DSU and if false, the signals are not added to the redistributor

Type

bool

Default value

0x0

affinity-width

A dotted quad indicating the bitwidth of fields at each affinity level

Type

string

Default value

4.8.8.8

allow-LPIEN-clear

Allow RW behaviour on `GICR_CTLR.LPIEN` instead of set once

Type

bool

Default value

0x1

chip-count

The total number of chips supported.

Type

int

Default value

0x10

chip-id

Chip ID when multichip operation is enabled.

Type

int

Default value

0x0

chip-select-affinity-level

Affinity level 2 or 3 can be used for chip select

Type

int

Default value

0x3

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

common-vPE-table-affinity

Affinity value list in the form of 'a.b.c.d, e.f.g.h, etc' given to the ITS(s) (where a.b.c.d corresponds to ITS0, e.f.g.h corresponds to ITS1 and so on). Under an affinity value the vPE configuration table is shared among redistributors where the level to be shared is defined by ITS-shared-vPE-table. This parameter is valid when has-gicv4.1 is true.

Type

string

Default value**consolidators**

Specify consolidators' base addresses, interrupt line counts and base interrupt IDs, in the form 'baseAddr0:itlineCount0:baseINTID0, baseAddr1:itlineCount1:baseINTID1, [etc]' (eg '0x3f100000:64:4096, 0x3f200000:64:4224'). The consolidators' count is inferred from the list (maximum of 4). If not specified, the component contains no consolidators.

Type

string

Default value**cross-chip-AMBA-is-ACE**

Indicates the AMBA protocol that the cross-chip interface uses. If true, the cross-chip interface uses the ACE5-Lite protocol. Otherwise, it uses the AXI5-Stream protocol. This parameter is meaningful only if enable-multichip-operation is set.

Type

bool

Default value

0x0

direct-lpi-support

Enable support for LPI operations through GICR registers

Type

bool

Default value

0x0

enable-local-cross-chip-addressing

If true, each distributor in a multichip system will use routing table address (ADDR) values programmed by local software, and will not be overwritten by the default chip. Otherwise, all the distributors will share the same routing table address values programmed in the default chip. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x0

enable-multiple-views-feature

If true, multiple view feature will provide multiple programming views which can be used by multiple hypervisors

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

extended-ppi-count

Number of extended PPI supported

Type

int

Default value

0x40

extended-spi-count

Number of extended SPI supported

Type

int

Default value

0x400

fmw-blktype-num

Number of stakeholder block types for FMU

Type

int

Default value

0x6

has-gicv4.1

Enable GICv4.1 functionality; when false the component is inactive.

Type

bool

Default value

0x1

has-two-security-states

If true, has two security states

Type

bool

Default value

0x1

max-cores-supported-by-GCI

GCI can support 1-64 cores. Maximum supported cores in GCI is configurable by device

Type

int

Default value

0x8

max-pe-on-chip

Maximum number of cores on any single chip. This will be used to identify the target chip and core.

Type

int

Default value

0x4

multichip-threaded-dgi

Enable sending multichip DGI messages in a separate thread. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x1

output_attributes

User-defined transform to be applied to bus attributes like MasterID, ExtendedID or UserFlags. Currently, only works for MPAM Attributes encoding into bus attributes.

Type

string

Default value

ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID,
ExtendedID[38]=MPAM_NS

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

prog-mpidr

Whether software or hardware can remove cores from a GIC configuration. (0: none, 1: prog - Secure software to remove cores during the boot up of a system. 2: strap - enables hardware to remove cores as GIC exits reset)

Type

int

Default value

0x0

redistributor-group

Redistributor grouping information with affinity as JSON : {"0":["0.0.0.0","0.0.0.1"],"1":["0.0.1.0","0.0.1.1"]} where RD with quad 0.0.0.0 and 0.0.0.1 belongs to RD group 0. All the RDs belong to one group when this parameter is not given.

Type

string

Default value

redistributor-group-file
File path to redistributor grouping information with affinity as JSON. The file uses the same format as "redistributor-group" parameter.

Type
string

Default value

redistributor-power-managed-by-pwrr
GIC-700 dedistributor power management is done by updating GICR_PWRR register

Type
bool

Default value
0x1

reg-base
GIC-700 base address

Type
int

Default value
0x2c010000

3.10.40 GIC720AE_Filter

GIC-720AE IRI implementation: Single/multi chip validation/filter component variant limited to 64 PE. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-989: IP revisions support

Revision	Quality level
N/A	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC720AE_Filter

This model has the following Iris instances:

Table 3-990: GIC720AE_Filter Iris instances

InstanceName	ComponentName
GIC720AE_Filter	GIC_IRI

InstanceName	ComponentName
GIC720AE_Filter.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC720AE_Filter.ITS0	GICv4InterruptTranslationService
GIC720AE_Filter.ITS0.bus_subordinate	PVBusSlave
GIC720AE_Filter.fmu	ExportTest.GIC720AE_Filterfmu
GIC720AE_Filter.fmu.pvbus_slave	PVBusSlave
GIC720AE_Filter.rd_0	GICv4RedistributorInternal
GIC720AE_Filter.rd_0_0	GICv4RedistributorInternal
GIC720AE_Filter.rd_0_0_0	GICv4RedistributorInternal
GIC720AE_Filter.rd_0_0_0_0	GICv3Redistributor
GIC720AE_Filter.rd_tl	GICv3Distributor

This model has the following MTI trace components:

Table 3-991: GIC720AE_Filter MTI instances

InstanceName	ComponentName
GIC720AE_Filter	GICv3IRI
GIC720AE_Filter.GICV3_ProtocolChecker	GICv3ProtocolChecker
GIC720AE_Filter.ITS0	GICv4InterruptTranslationService
GIC720AE_Filter.ITS0.bus_subordinate	PVBusSlave
GIC720AE_Filter.fmu	ExportTest.GIC720AE_Filterfmu
GIC720AE_Filter.fmu.pvbus_slave	PVBusSlave
GIC720AE_Filter.rd_0	GICv4RedistributorInternal
GIC720AE_Filter.rd_0_0	GICv4RedistributorInternal
GIC720AE_Filter.rd_0_0_0	GICv4RedistributorInternal
GIC720AE_Filter.rd_0_0_0_0	GICv3Redistributor
GIC720AE_Filter.rd_tl	GICv3Distributor

GIC720AE_Filter contains the following CADI targets:

- GIC720AE

About GIC720AE_Filter

The model has the same functionality as GIC700_Filter, but in addition supports the following AE-specific features:

- GIC FMU
- Multiple views

It has the same limitations as [3.10.38 GIC700_Filter](#) on page 3408.

Ports for GIC720AE_Filter

Table 3-992: Ports

Name	Protocol	Type	Description
apb_bus	PVBus	Slave	FMU signals
axi_stream_msi_s[32]	PVBus	Slave	AXI Stream Interface ports for MSI (message-signalled interrupt) translation, in GIC-720AE. Typically the SMMU's TCU connects to this port for MSI.
chip_id	Value	Slave	chip_id port for multichip operation
cpu_active_s[256]	Signal	Slave	CPUActive pins.
extended_ppi_in_<n>[64]	Signal	Slave	Extended private peripheral interrupts (ID1056-ID1119) for cpu <n>, where <n> is in the range 0-255.
extended_spi_in[1024]	Signal	Slave	Extended Shared peripheral interrupts.
fmu_cri	Signal	Master	Critical Interrupt
fmu_eri	Signal	Master	Error recovery Interrupt
icdrt_out	PVBus	Master	AXI Stream Interface. This is used only when local cross chip addressing is enabled at the moment
icrdt_in	PVBus	Slave	-
po_reset	Signal	Slave	Reset.
ppi_in_<n>[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
pvbus_filtermiss_m	PVBus	Master	Memory bus out. Transactions not filtered by the component.
pvbus_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvbus_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Reset.
spi_in[960]	Signal	Slave	Shared peripheral interrupts.
wake_request[256]	Signal	Master	Power management outputs.

Parameters for GIC720AE_Filter

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value**DS-behaviour**

GICD_CTLR.DS field behaviour: 0:RAZ/WI, 1:RAO/WI, 2:RW

Type

int

Default value

0x2

GICD_CTLR-DS-1-means-secure-only

If GICD_CTLR.DS=1, GICD supports a single security state which is secure if this is true, otherwise is non-secure.

Type

bool

Default value

0x0

GICD_TYPER2

GICD_TYPER2 value containing VID and VIL to define the width of vPEID for GICv4.1.

Type

int

Default value

0x0

IIDR

GICD_IIDR and GICR_IIDR value

Type

int

Default value

0x0

ITS-ID-bits

Number of interrupt bits supported by ITS.

Type

int

Default value

0x10

ITS-collection-ID-bits

Number of collection bits supported by ITS (optional parameter, 0 => 16bits support and GITS_TYPER.CIL=0)

Type

int

Default value

0x8

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

ITS-enable-itt-address-verification

If true, a transaction will be sent to ITT Address for verification.

Type

bool

Default value

0x0

ITS-hardware-collection-count

Number of hardware collections held exclusively in the ITS

Type

int

Default value

0x0

ITS-shared-vPE-table

Number of affinity levels to which the vPE configuration table is shared. This parameter is valid when has-gicv4.1 is true.

Type

int

Default value

0x0

ITS-vmovp-bit

Device supports software issuing a VMOVP to only one of the ITSs that has a mapping for a vPE. The device itself ensures synchronization of the VMOVP command across all ITSs that have mapping for that vPE.

Type

bool

Default value

0x0

PPI-count

Selects the number of PPI available for each PE as 8(id22-27,29,30), 12(id 20-31) or 16(id 16-31)

Type

int

Default value

0x10

SPI-blocks

Number of SPI blocks supported by the IRI, each block contains 32 SPIs

Type

int

Default value

0x1e

add-output-cpu-wake-request-signal-from-redistributor

if true, the redistributor will have the output signal cpu_wake_request from GIC to DSU and if false, the signals are not added to the redistributor

Type

bool

Default value

0x0

affinity-width

A dotted quad indicating the bitwidth of fields at each affinity level

Type

string

Default value

4.8.8.8

allow-LPIEN-clear

Allow RW behaviour on GICR_CTLR.LPIEN instead of set once

Type

bool

Default value

0x1

chip-count

The total number of chips supported.

Type

int

Default value

0x10

chip-id

Chip ID when multichip operation is enabled.

Type

int

Default value

0x0

chip-select-affinity-level

Affinity level 2 or 3 can be used for chip select

Type

int

Default value

0x3

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

common-vPE-table-affinity

Affinity value list in the form of 'a.b.c.d, e.f.g.h, etc' given to the ITS(s) (where a.b.c.d corresponds to ITS0, e.f.g.h corresponds to ITS1 and so on). Under an affinity value the vPE configuration table is shared among redistributors where the level to be shared is defined by ITS-shared-vPE-table. This parameter is valid when has-gicv4.1 is true.

Type

string

Default value

consolidators

Specify consolidators' base addresses, interrupt line counts and base interrupt IDs, in the form 'baseAddr0:itlineCount0:baseINTID0, baseAddr1:itlineCount1:baseINTID1, [etc]' (eg '0x3f100000:64:4096, 0x3f200000:64:4224'). The consolidators' count is inferred from the list (maximum of 4). If not specified, the component contains no consolidators.

Type

string

Default value

cross-chip-AMBA-is-ACE

Indicates the AMBA protocol that the cross-chip interface uses. If true, the cross-chip interface uses the ACE5-Lite protocol. Otherwise, it uses the AXI5-Stream protocol. This parameter is meaningful only if enable-multichip-operation is set.

Type

bool

Default value

0x0

direct-lpi-support

Enable support for LPI operations through GICR registers

Type

bool

Default value

0x0

enable-local-cross-chip-addressing

If true, each distributor in a multichip system will use routing table address (ADDR) values programmed by local software, and will not be overwritten by the default chip. Otherwise, all the distributors will share the same routing table address values programmed in the default chip. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x0

enable-multiple-views-feature

If true, multiple view feature will provide multiple programming views which can be used by multiple hypervisors

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

extended-ppi-count

Number of extended PPI supported

Type

int

Default value

0x40

extended-spi-count

Number of extended SPI supported

Type

int

Default value

0x400

fmv-blktype-num

Number of stakeholder block types for FMU

Type

int

Default value

0x6

has-gicv4.1

Enable GICv4.1 functionality; when false the component is inactive.

Type

bool

Default value

0x1

has-two-security-states

If true, has two security states

Type

bool

Default value

0x1

max-cores-supported-by-GCI

GCI can support 1-64 cores. Maximum supported cores in GCI is configurable by device

Type

int

Default value

0x8

max-pe-on-chip

Maximum number of cores on any single chip. This will be used to identify the target chip and core.

Type

int

Default value

0x4

multichip-threaded-dgi

Enable sending multichip DGI messages in a separate thread. This parameter is valid only if enable-multichip-operation is set.

Type

bool

Default value

0x1

output_attributes

User-defined transform to be applied to bus attributes like MasterID, ExtendedID or UserFlags. Currently, only works for MPAM Attributes encoding into bus attributes.

Type

string

Default value

ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID,
ExtendedID[38]=MPAM_NS

print-memory-map

Print memory map to stdout

Type

bool

Default value

0x0

prog-mpidr

Whether software or hardware can remove cores from a GIC configuration. (0: none, 1: prog - Secure software to remove cores during the boot up of a system. 2: strap - enables hardware to remove cores as GIC exits reset)

Type

int

Default value

0x0

redistributor-group

Redistributor grouping information with affinity as JSON : {"0":["0.0.0.0","0.0.0.1"],"1":["0.0.1.0","0.0.1.1"]} where RD with quad 0.0.0.0 and 0.0.0.1 belongs to RD group 0. All the RDs belong to one group when this parameter is not given.

Type

string

Default value**redistributor-group-file**

File path to redistributor grouping information with affinity as JSON. The file uses the same format as "redistributor-group" parameter.

Type

string

Default value**redistributor-power-managed-by-pwrr**

GIC-700 dedistributor power management is done by updating GICR_PWRR register

Type

bool

Default value

0x1

reg-base

GIC-700 base address

Type

int

Default value

0x2c010000

3.10.41 GIC_400

This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-993: IP revisions support

Revision	Quality level
r0p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for GIC_400

This model has the following Iris instances:

Table 3-994: GIC_400 Iris instances

InstanceName	ComponentName
GIC_400	GIC_400
GIC_400.vgic_bus_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-995: GIC_400 MTI instances

InstanceName	ComponentName
GIC_400	GIC_400
GIC_400.vgic_bus_slave	PVBusSlave

GIC_400 contains the following CADI targets:

- GIC_400

About GIC_400

This component is a wrapper that permits easier configuration of the v7_VGIC component that supports parameterized configuration.

The GIC-400 has several memory-mapped interfaces at the same address. The processor that is communicating with the GIC-400 banks them. The GIC-400 must be able to distinguish from which processor a transaction originates. In the hardware, the AUSER fields on AXI supply this information to the GIC-400. In the model, there is no exact equivalent to this field. However, each transaction has a `master_id` that the model can use to identify the originating processor.

Arm clusters assign the `master_id` as follows:

- Bits[31:16]: SBZ, which the GIC-400 ignores.
- Bits[5:2]: CLUSTERID.
- Bits[1:0]: `cpu_id` within cluster.

CLUSTERID is the 4-bit field that either a parameter on the processor sets or a value on the `clusterid` port drives. CPUID is the core number within the cluster. CLUSTERID appears in the CP15 register space as part of the MPIDR register.

The Arm® architecture suggests that each cluster in the system is given a different CLUSTERID. This distinction is essential for the VGIC to identify the cluster. The parameters in the GIC-400 component permit it to construct the map of `master_id` to interface number.

Processor interfaces that the GIC-400 supports have these parameters:

- `interfaceN.cluster_id`.
- `interfaceN.core_id`.
- `interfaceN.inout_port_number_to_use`.

N is the interface number (0-7). The `cluster_id` and `core_id` tell the GIC-400 to map that cluster or core combination to interface N.

In using `inout_port_number_to_use`, the GIC-400 has some input and output ports that pair with a particular processor interface. For example:

- The `irqcpu[]` pin wires to the `irq` port of the corresponding processor.
- The `cntpnsirq` pin from the processor wires to a `cntpnsirq[]` pin on GIC-400 to transport a *Private Peripheral Interrupt* (PPI) from the processor to the GIC-400.

The `interfaceN.inout_port_number_to_use` parameter supports clusters that can have variable numbers of cores. It tells the GIC-400 that to send to or receive a signal from the processor that is attached to interface N, it must use these pins:

- `irqout[interfaceN.inout_port_number_to_use]`.
- `fiqout[interfaceN.inout_port_number_to_use]`.
- `virqout[interfaceN.inout_port_number_to_use]`.
- `vfiqout[interfaceN.inout_port_number_to_use]`.
- `legacyirq[interfaceN.inout_port_number_to_use]`.
- `cntpnsirq[interfaceN.inout_port_number_to_use]`.
- `cntpsirq[interfaceN.inout_port_number_to_use]`.
- `legacyfiq[interfaceN.inout_port_number_to_use]`.
- `cntvirq[interfaceN.inout_port_number_to_use]`.
- `cnthpirq[interfaceN.inout_port_number_to_use]`.
- ...

`legacyirq` and `legacyfiq` are not signals from the processor but are signals into the GIC-400 from the legacy interrupt system. They are wired to PPIs. If the control registers of the GIC-400 are set up in particular ways, they can also bypass the GIC-400. See [ARM Generic Interrupt Controller Architecture version 2.0 Architecture Specification](#) for more information.

The fabric between the clusters and the GIC might remap the `master_id` of a transaction. If so, then the GIC might lose the ability to identify the originating processor. The fabrics that ARM ships in Fast Models perform no such transformation.

The comparison that the GIC-400 performs on the `master_id` is only on the bottom 6 bits of the `master_id`. It ignores the rest. If you are writing your own fabric and do not properly propagate the `master_id` or transform it, the GIC-400 might not be able to identify the processor. The source code for the GIC_400 component can be examined to see how it might be adapted for it to understand different `master_id` schemes.

Differences between the model and the RTL

The GIC-400 model has these limitations:

- Reads and writes to GICD_ISACTIVERn/GICD_ICACTIVERn/GICD_ISPENDRn/GICD_ICPENDRn might not work as expected unless there is a configured target in GICD_ICFGRm.
- Some of the interaction of GICD_CTLR.EnableGrpX and level sensitive interrupts might not work correctly.
- It does not model the signals nIRQOUT/nFIQOUT.
- It models interrupts with positive logic, rather than the negative logic that the hardware uses. Hence, the signal pins omit the “n” prefix in their names.

Ports for GIC_400

Table 3-996: Ports

Name	Protocol	Type	Description
<code>cfgsdisable</code>	Signal	Slave	Disable write access to some GIC registers.
<code>cnthpirq[8]</code>	Signal	Slave	Secure physical timer event. PPI interrupt id 26.
<code>cntpnsirq[8]</code>	Signal	Slave	Non-secure physical timer event. PPI interrupt id 30.
<code>cntpsirq[8]</code>	Signal	Slave	Secure physical timer event. PPI interrupt id 29.
<code>cntvirq[8]</code>	Signal	Slave	Virtual timer event. PPI interrupt id 27.
<code>fiqcpu[8]</code>	Signal	Master	FIQ signal to the corresponding processor.
<code>fiqout[8]</code>	Signal	Master	FIQOUT signal to the corresponding processor.
<code>irqcpu[8]</code>	Signal	Master	IRQ signal to the corresponding processor.
<code>irqout[8]</code>	Signal	Master	IRQOUT signal to the corresponding processor.
<code>irqs[480]</code>	Signal	Slave	Interrupt request input lines for the GIC.
<code>legacyfiq[8]</code>	Signal	Slave	Signal into the GIC-400 from the legacy interrupt system. PPI interrupt id 28.
<code>legacyirq[8]</code>	Signal	Slave	Signal into the GIC-400 from the legacy interrupt system. PPI interrupt id 31.
<code>pvbus_s</code>	PVBus	Slave	Handles incoming transactions from PVBus masters.
<code>reset_signal</code>	Signal	Slave	Reset signal input.
<code>vfiqcpu[8]</code>	Signal	Master	Virtual FIQ signal to the processor.
<code>virqcpu[8]</code>	Signal	Master	Virtual IRQ signal to the processor.

Parameters for GIC_400

NUM_CPUS

Number of interfaces to support

Type

int

Default value

0x1

NUM_SPIS

Number of interrupt pins

Type

int

Default value

0xe0

enable_log_errors

Type

bool

Default value

0x0

enable_log_fatal

Type

bool

Default value

0x0

enable_log_warnings

Type

bool

Default value

0x0

interface0.cluster_id

The CLUSTERID of the interface you want to appear as interface0 in the VGIC.

Type

int

Default value

0x0

interface0.core_id

The core id of interface0 in the cluster

Type

int

Default value

0x0

interface0.inout_port_number_to_use

Which ppiN port is used for this interface.

Type

int

Default value

0x0

interface1.cluster_id

The CLUSTERID of the interface you want to appear as interface1 in the VGIC.

Type

int

Default value

0x0

interface1.core_id

The core id of interface1 in the cluster

Type

int

Default value

0x0

interface1.inout_port_number_to_use

Which ppiN port is used for this interface.

Type

int

Default value

0x1

interface2.cluster_id

The CLUSTERID of the interface you want to appear as interface2 in the VGIC.

Type

int

Default value

0x0

interface2.core_id

The core id of interface2 in the cluster

Type

int

Default value

0x0

interface2.inout_port_number_to_use

Which ppiN port is used for this interface.

Type

int

Default value

0x2

interface3.cluster_id

The CLUSTERID of the interface you want to appear as interface3 in the VGIC.

Type

int

Default value

0x0

interface3.core_id

The core id of interface3 in the cluster

Type

int

Default value

0x0

interface3.inout_port_number_to_use

Which ppiN port is used for this interface.

Type

int

Default value

0x3

interface4.cluster_id

The CLUSTERID of the interface you want to appear as interface4 in the VGIC.

Type

int

Default value

0x0

interface4.core_id

The core id of interface4 in the cluster

Type

int

Default value

0x0

interface4.inout_port_number_to_use

Which ppiN port is used for this interface.

Type

int

Default value

0x4

interface5.cluster_id

The CLUSTERID of the interface you want to appear as interface5 in the VGIC.

Type

int

Default value

0x0

interface5.core_id

The core id of interface5 in the cluster

Type

int

Default value

0x0

interface5.inout_port_number_to_use

Which ppiN port is used for this interface.

Type

int

Default value

0x5

interface6.cluster_id

The CLUSTERID of the interface you want to appear as interface6 in the VGIC.

Type

int

Default value

0x0

interface6.core_id

The core id of interface6 in the cluster

Type

int

Default value

0x0

interface6.inout_port_number_to_use

Which ppiN port is used for this interface.

Type

int

Default value

0x6

interface7.cluster_id

The CLUSTERID of the interface you want to appear as interface7 in the VGIC.

Type

int

Default value

0x0

interface7.core_id

The core id of interface7 in the cluster

Type

int

Default value

0x0

interface7.inout_port_number_to_use

Which ppiN port is used for this interface.

Type

int

Default value

0x7

3.10.42 GIC_IRI

GIC metacomponent for redistribution of interrupts. Contains a Distributor and a configurable number of ITSs and Redistributors. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-997: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters removed:

- `has-gicv3`

Iris and MTI instances for GIC_IRI

This model has the following Iris instances:

Table 3-998: GIC_IRI Iris instances

InstanceName	ComponentName
GIC_IRI	GIC_IRI
GIC_IRI.rd_0	GICv3RedistributorInternal
GIC_IRI.rd_0_0	GICv3RedistributorInternal
GIC_IRI.rd_0_0_0	GICv3RedistributorInternal
GIC_IRI.rd_0_0_0_0	GICv3Redistributor
GIC_IRI.rd_tl	GICv3Distributor

This model has the following MTI trace components:

Table 3-999: GIC_IRI MTI instances

InstanceName	ComponentName
GIC_IRI	GICv3IRI
GIC_IRI.rd_0	GICv3RedistributorInternal
GIC_IRI.rd_0_0	GICv3RedistributorInternal
GIC_IRI.rd_0_0_0	GICv3RedistributorInternal
GIC_IRI.rd_0_0_0_0	GICv3Redistributor
GIC_IRI.rd_tl	GICv3Distributor

GIC_IRI contains the following CADI targets:

- GIC_IRI

About GIC_IRI

The GIC_IRI has one slave PVBUS interface and one master PVBUS interface. It behaves in a similar manner to a GICv3-compatible device, with the slave interface, `pvbuss_s`, granting access to the register banks used by the configuration and operation of MSIs and the master interface, `pvbuss_m`, issuing transactions that are required by the ITS and the redistributors in LPI-related operations. All transactions that are routed to the slave port terminate in the component. Accesses to unmapped space are RAZ/WI.

An instance of GICv3 requires a small set of parameters to be configured to be useful. For example:

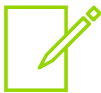
```
gic_iri : GIC_IRI(
    "reg-base" = 0xF0020000, //Base address for GICD_* REGISTERS, 64K aligned
    "CPU-affinities" = "0.0.0.0, 0.0.1.0, 0.0.1.1",
    //A comma-separated list of affinity addresses corresponding to
    //cpu affinities in the system
    "reg-base-per-redistributor"="0.0.0.0=0xF0040000,0.0.1.0=0xF0060000,
0.0.0.0=0xF0080000",
    //Base addresses for each redistributor in a comma-separated list of
    //affinity=address
);
```

To use LPIs, an ITS must be configured. A minimal configuration might consist of, for example:

```
"ITS-count" = 1, //The number of ITSs in the IRI. Defaults to zero.
"ITS0-base" = 0xF0100000,
"GITS_BASER0-type" = 1, //Type 1 is Devices. A device table is always needed.
"GITS_BASER2-type" = 4, //Type 4 is Collections.
//A collection table is needed if GITS_TYPER.HCC is 0.
```

To use GICv4 functionality, one or more ITSs must be configured, as shown in the previous example. In addition, the following parameters are required:

```
"virtual-lpi-support"=true,
"GITS_BASER4-type"=2 //Type 2 is Virtual PEs.
//Such a table is needed for GICv4 functionality.
```



Note

- Set the `FASTSIM_GIC_MEMORY_MAP` environment variable to 1 to print to `stderr` the memory map of any GICv3 or later models that are included in the platform being run.
- Set the `GICD_ITARGETSR-RAZWI` parameter to true for legacy, GICv2-style routing, where interrupts are routed to the first processor in the system.

Ports for GIC_IRI

Table 3-1000: Ports

Name	Protocol	Type	Description
cfgsdisable	Signal	Slave	Disable some SPIs signal.
extended_ppi_in_<n>[64]	Signal	Slave	Extended private peripheral interrupts (ID1056-ID1119) for cpu <n>, where <n> is in the range 0-255.
extended_spi_in[1024]	Signal	Slave	Extended Shared peripheral interrupts.
msi_error_interrupt	Signal	Master	When report of MSI error allowed through interrupt, loop-back this signal to relevant IRQ input signal
po_reset	Signal	Slave	Resets.
ppi_in_<n>[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
pvb_m	PVBus	Master	Memory bus out: transactions generated by the IRI.
pvb_s	PVBus	Slave	Memory bus in: memory-mapped register accesses are accepted on this interface.

Name	Protocol	Type	Description
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Resets.
spi_in[988]	Signal	Slave	Shared peripheral interrupts.
wake_request[256]	Signal	Master	Power management outputs.
wire_to_msi_in_0[1024]	Signal	Slave	Wire-to-MSI interrupts for architectural consolidator 0.
wire_to_msi_in_1[1024]	Signal	Slave	Wire-to-MSI interrupts for architectural consolidator 1.
wire_to_msi_in_2[1024]	Signal	Slave	Wire-to-MSI interrupts for architectural consolidator 2.
wire_to_msi_in_3[1024]	Signal	Slave	Wire-to-MSI interrupts for architectural consolidator 3.

Parameters for GIC_IRI

A3-affinity-supported

Device supports affinity level 3 values that are non-zero.

Type

bool

Default value

0x0

ARE-fixed-to-one

GICv2 compatibility is not supported and GICD_CTLR.ARE_* is always one

Type

bool

Default value

0x0

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value**DPG-ARE-only**

Limit application of DPG bits to interrupt groups for which ARE=1

Type

bool

Default value

0x0

DPG-bits-implemented

Enable implementation of interrupt group participation bits or DPG bits in GICR_CTLR

Type

bool

Default value

0x0

DS-fixed-to-zero

Enable/disable support of single security state

Type

bool

Default value

0x0

GICD-alias

In GICv2 mode: the base address for a 4k page alias of the first 4k of the Distributor page, in GICv3 mode: the base address of a 64KB page containing message based SPI signalling register aliases(0:Disabled)

Type

int

Default value

0x0

GICD-legacy-registers-as-reserved

When ARE is RAO/WI, makes superfluous registers in GICD reserved (including for the purpose of STATUSR updates)

Type

bool

Default value

0x0

GICD_CTLR.DS-1-means-secure-only

If GICD_CTLR.DS=1, GICD supports a single security state which is secure if this is true, otherwise is non-secure.

Type

bool

Default value

0x0

GICD_ITARGETSR-RAZWI

If true, the GICD_ITARGETS registers are RAZ/WI

Type

bool

Default value

0x0

GICD_PIDR

The value for the GICD_PIDR registers, if non-zero. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

GICD_TYPER2

GICD_TYPER2 value containing VID and VIL to define the width of vPEID for GICv4.1.

Type

int

Default value

0x0

GICR-clear-enable-supported

When true, this sets the value of the RO bit GICR_CTLR.CES with the value of the parameter allow-LPIEN-clear, making it visible to software

Type

bool

Default value

0x0

GICR-invalidate-registers-implemented

When true, the registers GICR_INVLPIR, GICR_INVALLR and GICR_SYNCR are implemented

Type

bool

Default value

0x0

GICR_PIDR

The value for the GICR_PIDR registers, if non-zero. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

GICR_PROPBASER-read-only

GICR_PROPBASER register is read-only.

Type

bool

Default value

0x0

GICR_PROPBASER-reset-value

Value of GICR_PROPBASER on reset.

Type

int

Default value

0x0

GITS_BASER0-entry-bytes

Number of bytes required per entry for GITS_BASER0 register.

Type

int

Default value

0x8

GITS_BASER0-indirect-RAZ

Indirect field for GITS_BASER0 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER0-type

Type field for GITS_BASER0 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER1-entry-bytes

Number of bytes required per entry for GITS_BASER1 register.

Type

int

Default value

0x8

GITS_BASER1-indirect-RAZ

Indirect field for GITS_BASER1 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER1-type

Type field for GITS_BASER1 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER2-entry-bytes

Number of bytes required per entry for GITS_BASER2 register.

Type

int

Default value

0x8

GITS_BASER2-indirect-RAZ

Indirect field for GITS_BASER2 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER2-type

Type field for GITS_BASER2 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER3-entry-bytes

Number of bytes required per entry for GITS_BASER3 register.

Type

int

Default value

0x8

GITS_BASER3-indirect-RAZ

Indirect field for GITS_BASER3 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER3-type

Type field for GITS_BASER3 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER4-entry-bytes

Number of bytes required per entry for GITS_BASER4 register.

Type

int

Default value

0x8

GITS_BASER4-indirect-RAZ

Indirect field for GITS_BASER4 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER4-type

Type field for GITS_BASER4 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER5-entry-bytes

Number of bytes required per entry for GITS_BASER5 register.

Type

int

Default value

0x8

GITS_BASER5-indirect-RAZ

Indirect field for GITS_BASER5 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER5-type

Type field for GITS_BASER5 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER6-entry-bytes

Number of bytes required per entry for GITS_BASER6 register.

Type

int

Default value

0x8

GITS_BASER6-indirect-RAZ

Indirect field for GITS_BASER6 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER6-type

Type field for GITS_BASER6 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER7-entry-bytes

Number of bytes required per entry for GITS_BASER7 register.

Type

int

Default value

0x8

GITS_BASER7-indirect-RAZ

Indirect field for GITS_BASER7 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER7-type

Type field for GITS_BASER7 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_PIDR

The value for the GITS_PIDR registers, if non-zero. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

ICFGR-PPI-mask

Mask for writes to ICFGR registers that configure PPIs

Type

int

Default value

0xaaaaaaaa

ICFGR-PPI-reset

Reset value for ICFGR registers that configure PPIs

Type

int

Default value

0x0

ICFGR-SGI-mask

Mask for writes to ICFGR registers that configure SGIs

Type

int

Default value

0x0

ICFGR-SGI-reset

Reset value for ICFGR registers that configure SGIs

Type

int

Default value

0xaaaaaaaa

ICFGR-SPI-mask

Mask for writes to ICFGR registers that configure SPIs

Type

int

Default value

0xaaaaaaaa

ICFGR-SPI-reset

Reset value for ICFGR registers that configure SPIs

Type

int

Default value

0x0

ICFGR-rsvd-bit

If ARE=0, the value of reserved bits i.e. bit 0,2,4..30 of ICFGRn for n>0

Type

bool

Default value

0x0

IGROUP-PPI-mask

Mask for writes to PPI bits in IGROUP registers

Type

int

Default value

0xffff

IGROUP-PPI-reset

Reset value for SGI bits in IGROUP registers

Type

int

Default value

0x0

IGROUP-SGI-mask

Mask for writes to SGI bits in IGROUP registers

Type

int

Default value

0xffff

IGROUP-SGI-reset

Reset value for SGI bits in IGROUP registers

Type

int

Default value

0x0

IIDR

GICD_IIDR and GICR_IIDR value

Type

int

Default value

0x0

IRI-ID-bits

Number of bits used to represent interrupts IDs in the Distributor and Redistributors, forced to 10 when none of LPIs, extended SPLs or extended PPIs is supported

Type

int

Default value

0x10

IROUTER-IRM-RAZ-WI

GICD_IROUTERn.InterruptRoutingMode is RAZ/WI

Type

bool

Default value

0x0

ITS-BASER-force-page-alignment

Force alignment of address written to a GITS_BASER register to the page size configured

Type

bool

Default value

0x1

ITS-ID-bits

Number of interrupt bits supported by ITS.

Type

int

Default value

0x10

ITS-MOVALL-update-collections

Whether MOVALL command updates the collection entires

Type

bool

Default value

0x0

ITS-TRANSLATE64R

Add an implementation specific register at 0x10008 supporting 64 bit TRANSLATER (dev[63:32], interrupt[31:0])

Type

bool

Default value

0x0

ITS-cache-invalidate-on-disable

Sets the RO bit GITS_TYPER.INV. When true, after the following sequence: 1) GITS_CTLR.Enabled written 1-->0, 2) GITS_CTLR.Quirescent observed as 1, 3) GITS_BASER<n>.Valid written 1-->0, there is no cached information from the ITS memory structure pointed to by GITS_BASER<n>.

Type

bool

Default value

0x0

ITS-collection-ID-bits

Number of collection bits supported by ITS (optional parameter, 0 => 16bits support and GITS_TYPER.CIL=0)

Type

int

Default value

0x0

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x0

ITS-cumulative-collection-tables

When true, the supported amount of collections is the sum of GITS_TYPER.HCC and the number of collections supported in memory, otherwise, simply the number supported in memory only. Irrelevant when HCC=0

Type

bool

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

ITS-enable-itt-address-verification

If true, a transaction will be sent to ITT Address for verification.

Type

bool

Default value

0x0

ITS-entry-size

Number of bytes required to store each entry in the ITT tables.

Type

int

Default value

0x8

ITS-hardware-collection-count

Number of hardware collections held exclusively in the ITS

Type

int

Default value

0x0

ITS-legacy-iidr-typer-offset

Put the GITS_IIDR and GITS_TYPER registers at their older offset of 0x8 and 0x4 respectively

Type

bool

Default value

0x0

ITS-shared-vpe-table

Number of affinity levels to which the vPE configuration table is shared. This parameter is valid when has-gicv4.1 is true.

Type

int

Default value

0x0

ITS-threaded-command-queue

Enable execution of ITS commands in a separate thread which is sometimes required for cosimulation

Type

bool

Default value

0x1

ITS-use-physical-target-addresses

Use physical hardware addresses for targets in ITS commands -- must be true for distributed implementations

Type

bool

Default value

0x1

ITS-vmovp-bit

Device supports software issuing a VMOVP to only one of the ITSs that has a mapping for a vPE. The device itself ensures synchronization of the VMOVP command across all ITSs that have mapping for that vPE.

Type

bool

Default value

0x0

ITS0-base

Register base address for ITS0 (automatic if 0).

Type

int

Default value

0x0

ITS1-base

Register base address for ITS1 (automatic if 0).

Type

int

Default value

0x0

ITS2-base

Register base address for ITS2 (automatic if 0).

Type

int

Default value

0x0

ITS3-base

Register base address for ITS3 (automatic if 0).

Type

int

Default value

0x0

LPI-cache-check-data

Enable Cached LPI data against memory checking when available for cache type

Type

bool

Default value

0x0

LPI-cache-type

Cache type for LPIs, 0:No caching, 1:Full caching

Type

int

Default value

0x1

MSI_IIDR

Value returned in MSI_IIDR registers.

Type

int

Default value

0x0

MSI_NS-frame0-base

If non-zero, sets the base address used for non-secure MSI frame 0 registers.

Type

int

Default value

0x0

MSI_NS-frame0-max-SPI

Maximum SPI ID supported by non-secure MSI frame 0. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame0-min-SPI

Minimum SPI ID supported by non-secure MSI frame 0. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame1-base

If non-zero, sets the base address used for non-secure MSI frame 1 registers.

Type

int

Default value

0x0

MSI_NS-frame1-max-SPI

Maximum SPI ID supported by non-secure MSI frame 1. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame1-min-SPI

Minimum SPI ID supported by non-secure MSI frame 1. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame2-base

If non-zero, sets the base address used for non-secure MSI frame 2 registers.

Type

int

Default value

0x0

MSI_NS-frame2-max-SPI

Maximum SPI ID supported by non-secure MSI frame 2. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame2-min-SPI

Minimum SPI ID supported by non-secure MSI frame 2. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame3-base

If non-zero, sets the base address used for non-secure MSI frame 3 registers.

Type

int

Default value

0x0

MSI_NS-frame3-max-SPI

Maximum SPI ID supported by non-secure MSI frame 3. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame3-min-SPI

Minimum SPI ID supported by non-secure MSI frame 3. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame4-base

If non-zero, sets the base address used for non-secure MSI frame 4 registers.

Type

int

Default value

0x0

MSI_NS-frame4-max-SPI

Maximum SPI ID supported by non-secure MSI frame 4. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame4-min-SPI

Minimum SPI ID supported by non-secure MSI frame 4. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame5-base

If non-zero, sets the base address used for non-secure MSI frame 5 registers.

Type

int

Default value

0x0

MSI_NS-frame5-max-SPI

Maximum SPI ID supported by non-secure MSI frame 5. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame5-min-SPI

Minimum SPI ID supported by non-secure MSI frame 5. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame6-base

If non-zero, sets the base address used for non-secure MSI frame 6 registers.

Type

int

Default value

0x0

MSI_NS-frame6-max-SPI

Maximum SPI ID supported by non-secure MSI frame 6. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame6-min-SPI

Minimum SPI ID supported by non-secure MSI frame 6. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame7-base

If non-zero, sets the base address used for non-secure MSI frame 7 registers.

Type

int

Default value

0x0

MSI_NS-frame7-max-SPI

Maximum SPI ID supported by non-secure MSI frame 7. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame7-min-SPI

Minimum SPI ID supported by non-secure MSI frame 7. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_PIDR

The value for the MSI_PIDR registers, if non-zero and distributor supports GICv2m. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

MSI_S-frame0-base

If non-zero, sets the base address used for secure MSI frame 0 registers.

Type

int

Default value

0x0

MSI_S-frame0-max-SPI

Maximum SPI ID supported by secure MSI frame 0. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame0-min-SPI

Minimum SPI ID supported by secure MSI frame 0. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame1-base

If non-zero, sets the base address used for secure MSI frame 1 registers.

Type

int

Default value

0x0

MSI_S-frame1-max-SPI

Maximum SPI ID supported by secure MSI frame 1. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame1-min-SPI

Minimum SPI ID supported by secure MSI frame 1. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame2-base

If non-zero, sets the base address used for secure MSI frame 2 registers.

Type

int

Default value

0x0

MSI_S-frame2-max-SPI

Maximum SPI ID supported by secure MSI frame 2. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame2-min-SPI

Minimum SPI ID supported by secure MSI frame 2. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame3-base

If non-zero, sets the base address used for secure MSI frame 3 registers.

Type

int

Default value

0x0

MSI_S-frame3-max-SPI

Maximum SPI ID supported by secure MSI frame 3. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame3-min-SPI

Minimum SPI ID supported by secure MSI frame 3. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame4-base

If non-zero, sets the base address used for secure MSI frame 4 registers.

Type

int

Default value

0x0

MSI_S-frame4-max-SPI

Maximum SPI ID supported by secure MSI frame 4. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame4-min-SPI

Minimum SPI ID supported by secure MSI frame 4. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame5-base

If non-zero, sets the base address used for secure MSI frame 5 registers.

Type

int

Default value

0x0

MSI_S-frame5-max-SPI

Maximum SPI ID supported by secure MSI frame 5. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame5-min-SPI

Minimum SPI ID supported by secure MSI frame 5. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame6-base

If non-zero, sets the base address used for secure MSI frame 6 registers.

Type

int

Default value

0x0

MSI_S-frame6-max-SPI

Maximum SPI ID supported by secure MSI frame 6. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame6-min-SPI

Minimum SPI ID supported by secure MSI frame 6. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame7-base

If non-zero, sets the base address used for secure MSI frame 7 registers.

Type

int

Default value

0x0

MSI_S-frame7-max-SPI

Maximum SPI ID supported by secure MSI frame 7. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame7-min-SPI

Minimum SPI ID supported by secure MSI frame 7. Set to 0 to disable frame.

Type

int

Default value

0x0

PA_SIZE

Number of valid bits in physical address

Type

int

Default value

0x30

PPI-implemented-mask

Mask of PPIs that are implemented. One bit per PPI bit 0 == PPI 16 (first PPI). This will affect other masks.

Type

int

Default value

0xffff

SPI-count

Number of SPIs that are implemented.

Type

int

Default value

0xe0

SPI-message-based-support

Distributor supports message based signaling of SPI

Type

bool

Default value

0x1

SPI-unimplemented

A comma separated list of unimplemented SPIs ranges for sparse SPI definition(for ex: '35, 39-42, 73')

Type

string

Default value**STATUSR-implemented**

Determines whether the GICR_STATUSR register is implemented.

Type

bool

Default value

0x1

add-output-cpu-wake-request-signal-from-redistributor

if true, the redistributor will have the output signal cpu_wake_request from GIC to DSU and if false, the signals are not added to the redistributor

Type

bool

Default value

0x0

allow-LPIEN-clear

Allow RW behaviour on GICR_CTLR.LPIEN instead of set once

Type

bool

Default value

0x0

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

common-lpi-configuration

Describes which re-distributors share (and must be configured with the same) LPI configuration table as described in GICR_TYPER(0:All, 1:A.x.x.x, 2:A.B.x.x, 3:A.B.C.x

Type

int

Default value

0x0

common-vPE-table-affinity

Affinity value list in the form of 'a.b.c.d, e.f.g.h, etc' given to the ITS(s) (where a.b.c.d corresponds to ITS0, e.f.g.h corresponds to ITS1 and so on). Under an affinity value the vPE configuration table is shared among redistributors where the level to be shared is defined by ITS-shared-vPE-table. This parameter is valid when has-gicv4.1 is true.

Type

string

Default value**consolidators**

Specify consolidators' base addresses, interrupt line counts and base interrupt IDs, in the form 'baseAddr0:itlineCount0:baseINTID0, baseAddr1:itlineCount1:baseINTID1, [etc]' (eg '0x3f100000:64:4096, 0x3f200000:64:4224'). The consolidators' count is inferred from the list (maximum of 4). If not specified, the component contains no consolidators.

Type

string

Default value**delay-ITS-accesses**

Delay accesses from the ITS until GICR_SYNCRR is read.

Type

bool

Default value

0x1

delay-redistributor-accesses

Delay memory accesses from the redistributor until GICR_SYNCRR is read.

Type

bool

Default value

0x1

direct-lpi-support

Enable support for LPI operations through GICR registers

Type

bool

Default value

0x0

enable_protocol_checking

Enable/disable protocol checking at cpu interface

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

extended-ppi-count

Number of extended PPI supported

Type

int

Default value

0x0

extended-spi-count

Number of extended SPI supported

Type

int

Default value

0x0

fixed-routed-spis

Value of IROUTER[n] register in the form 'n=a.b.c.d, n=*'. The RM bit of IROUTER is 0 when n=a.b.c.d is used else 1 when n=* is used. n can be >= 32 and <= 1019

Type

string

Default value

gicr-icfgr-extended-count

Number of extended GICR_ICFGR registers supported

Type

int

Default value

0x4

gicv2-only

If true, when using the GICv3 model, pretend to be a GICv2 system

Type

bool

Default value

0x0

group-enables-control-doorbell

When true, GICR_VPENDBASER.{VGrp0En,VGrp1En} are cached to allow GIC to check group enables when virtual interrupt targeting this VCPU which is non-resident reaches Redistributor.

Type

bool

Default value

0x0

has-gicv4.1

Enable GICv4.1 functionality; when false the component is inactive.

Type

bool

Default value

0x0

has-two-security-states

If true, has two security states

Type

bool

Default value

0x1

has_VPENDBASER-dirty-flag-on-load

GICR_VPENDBASER.Dirty reflects transient loading state when valid=1

Type

bool

Default value

0x0

has_mpam

Enable MPAM support on ITS and RDs

Type

bool

Default value

0x0

has_nmi

Enable support for Non-maskable Interrupts (NMIs). (FEAT_GICv3_NMI)

Type

bool

Default value

0x0

ignore-generate-sgi-when-no-are

Ignore GenerateSGI packets coming from the CPU interface if both ARE_S and ARE_NS are 0

Type

bool

Default value

0x0

individual-doorbell-not-supported

For IRI with support of virtual interrupt, individual doorbell is not supported when true.

Type

bool

Default value

0x0

irouter-default-mask

Default Mask value for IROUTER[32..1019] register in the form 'a.b.c.d'

Type

string

Default value**irouter-default-reset**

Default Reset Value of IROUTER[32..1019] register in the form 'a.b.c.d' or *

Type

string

Default value**irouter-mask-values**

Mask Value of IROUTER[n] register in the form 'n=a.b.c.d'.n can be ≥ 32 and ≤ 1019

Type

string

Default value**irouter-reset-values**

Reset Value of IROUTER[n] register in the form 'n=a.b.c.d or n=*'.n can be ≥ 32 and ≤ 1019

Type

string

Default value**legacy-sgi-enable-rao**

Enables for SGI associated with an ARE=0 regime are RAO/WI

Type

bool

Default value

0x0

local-SEIs

Generate SEI to signal internal issues

Type

bool

Default value

0x0

local-VSEIs

Generate VSEI to signal internal issues

Type

bool

Default value

0x0

lockable-SPI-count

Number of SPIs that are locked down when CFGSDISABLE signal is asserted. Only applies for GICv2.

Type

int

Default value

0x0

monolithic

Indicate that the implementation is not distributed

Type

bool

Default value

0x0

mpam_max_partid

Maximum valid PARTID

Type

int

Default value

0xffff

mpam_max_pmg

Maximum valid PMG

Type

int

Default value

0xff

non-ARE-core-count

Maximum number of non-ARE cores; normally used to pass the cluster-level NUM_CORES parameter to the top-level redistributor.

Type

int

Default value

0x8

outer-cacheability-support

Allow configuration of outer cachability attributes in ITS and Redistributor

Type

bool

Default value

0x0

output_attributes

User-defined transform to be applied to bus attributes like MasterID, ExtendedID or UserFlags. Currently, only works for MPAM Attributes encoding into bus attributes.

Type

string

Default valueExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID,
ExtendedID[38]=MPAM_NS**print-memory-map**

Print memory map to stdout

Type

bool

Default value

0x0

priority-bits

Number of implemented priority bits

Type

int

Default value

0x5

processor-numbersSpecify processor numbers (as appears in GICR_TYPER) in the form 0.0.0.0=0,0.0.0.1=1 etc.)
If not specified, will number processors starting at 0.**Type**

string

Default value**redistributor-threaded-sync**Enable execution of redistributor delayed transactions in a separate thread which is
sometimes required for cosimulation**Type**

bool

Default value

0x1

reg-base

Base for decoding GICv3 registers.

Type

int

Default value

0x2c010000

reg-base-per-redistributor

Base address for each redistributor in the form '0.0.0.0=0x2c010000, 0.0.0.1=0x2c020000'. All redistributors must be specified and this overrides the reg-base parameter (except that reg-base will still be used for the top-level redistributor). If reg-base-per-redistributor-file is specified, this parameter is ignored.

Type

string

Default value**reg-base-per-redistributor-file**

Path to file containing the base address for each redistributor in the form '0.0.0.0=0x2c010000, 0.0.0.1=0x2c020000'. All redistributors must be specified and this overrides the reg-base parameter (except that reg-base will still be used for the top-level redistributor). If this parameter is specified, reg-base-per-redistributor parameter will be ignored even when it is given.

Type

string

Default value**report-MSI-error-via-statusr**

Report MSI error via GITS_STATUSR. (0:unsupported, 1:report by GITS_STATUSR, 2:report by GITS_STATUSR and interrupt as well).

Type

int

Default value

0x0

rme_default_mecid_nonsecure

Default MECID value for NON-SECURE PAS

Type

int

Default value

0x0

sgi-range-selector-support

Device has support for the Range Selector feature for SGI

Type

bool

Default value

0x0

single-set-support

When true, forces redistributors to recall interrupts with a clear rather than issue a second Set command

Type

bool

Default value

0x0

supports-shareability

Device supports shareability attributes on outgoing memory bus (i.e. is modelling an ACElite port rather than an AXI4 port).

Type

bool

Default value

0x1

trace-speculative-lpi-property-update

Trace LPI property updates performed on speculative accesses (useful for debugging LPI)

Type

bool

Default value

0x0

vPE-table-entry-size-in-doubleword

The size of one entry of a vPE configuration table in double word. The value decremented by one is shown at GICR_VPROPBASER.Entry_Size. Current model mandates the minimum entry size to be 4 doublewords. When lower value is given, it is truncated to 4.

Type

int

Default value

0x5

virtual-lpi-support

GICv4 Virtual LPIs and Direct injection of Virtual LPIs supported

Type

bool

Default value

0x0

virtual-priority-bits

Number of implemented virtual priority bits

Type

int

Default value

0x5

wakeup-on-reset

Go against specification and start redistributors in woken-up state at reset. This allows software that was written for previous versions of the GICv3 specification to work correctly. This should not be used for production code or when the distributor is used separately from the core fast model.

Type

bool

Default value

0x0

3.10.43 GIC_IRI_Filter

GIC metacomponent for redistribution of interrupts. Contains a Distributor and a configurable number of ITSs and Redistributors. Validation only version. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1001: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters removed:

- has-gicv3

Iris and MTI instances for GIC_IRI_Filter

This model has the following Iris instances:

Table 3-1002: GIC_IRI_Filter Iris instances

InstanceName	ComponentName
GIC_IRI_Filter	GIC_IRI
GIC_IRI_Filter.rd_0	GICv3RedistributorInternal
GIC_IRI_Filter.rd_0_0	GICv3RedistributorInternal
GIC_IRI_Filter.rd_0_0_0	GICv3RedistributorInternal
GIC_IRI_Filter.rd_0_0_0_0	GICv3Redistributor

InstanceName	ComponentName
GIC_IRI_Filter.rd_tl	GICv3Distributor

This model has the following MTI trace components:

Table 3-1003: GIC_IRI_Filter MTI instances

InstanceName	ComponentName
GIC_IRI_Filter	GICv3IRI
GIC_IRI_Filter.rd_0	GICv3RedistributorInternal
GIC_IRI_Filter.rd_0_0	GICv3RedistributorInternal
GIC_IRI_Filter.rd_0_0_0	GICv3RedistributorInternal
GIC_IRI_Filter.rd_0_0_0_0	GICv3Redistributor
GIC_IRI_Filter.rd_tl	GICv3Distributor

GIC_IRI_Filter contains the following CADI targets:

- GIC_IRI

About GIC_IRI_Filter

The GIC_IRI_Filter has similar behavior to the GIC_IRI, except for the slave interface. Any transaction accessing a 4KB page that is not used by the GIC, as configurable through the parameters, is forwarded to the `pvbus_filtermiss_m` port, which is only present in this variant.



Note

- Set the `FASTSIM_GIC_MEMORY_MAP` environment variable to 1 to print to `stderr` the memory map of any GICv3 or later models that are included in the platform being run.
- Set the `GICD_ITARGETSR-RAZWI` parameter to true for legacy, GICv2-style routing, where interrupts are routed to the first processor in the system

Ports for GIC_IRI_Filter

Table 3-1004: Ports

Name	Protocol	Type	Description
<code>cfgsdisable</code>	Signal	Slave	Disable some SPIs signal.
<code>extended_ppi_in_<n>[64]</code>	Signal	Slave	Extended private peripheral interrupts (ID1056-ID1119) for cpu <n>, where <n> is in the range 0-255.
<code>extended_spi_in[1024]</code>	Signal	Slave	Extended Shared peripheral interrupts.
<code>msi_error_interrupt</code>	Signal	Master	When report of MSI error allowed through interrupt, loop-back this signal to relevant IRQ input signal
<code>po_reset</code>	Signal	Slave	Resets.
<code>ppi_in_<n>[16]</code>	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu <n>, where n is in the range 0-255.
<code>pvbus_filtermiss_m</code>	PVBus	Master	Passthrough for accesses to pages not used by the GIC IRI.
<code>pvbus_m</code>	PVBus	Master	Memory bus for transactions generated by the GIC.

Name	Protocol	Type	Description
pvbus_s	PVBus	Slave	Memory bus in.
redistributor_m[256]	GICv3Comms	Master	Input from and output to CPU interface.
reset	Signal	Slave	Resets.
spi_in[988]	Signal	Slave	Shared peripheral interrupts.
wake_request[256]	Signal	Master	Power management outputs.
wire_to_msi_in_0[1024]	Signal	Slave	Wire-to-MSI interrupts for architectural consolidator 0.
wire_to_msi_in_1[1024]	Signal	Slave	Wire-to-MSI interrupts for architectural consolidator 1.
wire_to_msi_in_2[1024]	Signal	Slave	Wire-to-MSI interrupts for architectural consolidator 2.
wire_to_msi_in_3[1024]	Signal	Slave	Wire-to-MSI interrupts for architectural consolidator 3.

Parameters for GIC_IRI_Filter

A3-affinity-supported

Device supports affinity level 3 values that are non-zero.

Type

bool

Default value

0x0

ARE-fixed-to-one

GICv2 compatibility is not supported and GICD_CTLR.ARE_* is always one

Type

bool

Default value

0x0

CPU-affinities

A comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If CPU-affinities-file is specified, this parameter is ignored.

Type

string

Default value

0.0.0.0

CPU-affinities-file

A file containing comma separated list of dotted quads containing the affinities of all PEs connected to this IRI. If this parameter is specified, CPU-affinities parameter will be ignored even when it is given.

Type

string

Default value**DPG-ARE-only**

Limit application of DPG bits to interrupt groups for which ARE=1

Type

bool

Default value

0x0

DPG-bits-implemented

Enable implementation of interrupt group participation bits or DPG bits in GICR_CTLR

Type

bool

Default value

0x0

DS-fixed-to-zero

Enable/disable support of single security state

Type

bool

Default value

0x0

GICD-alias

In GICv2 mode: the base address for a 4k page alias of the first 4k of the Distributor page, in GICv3 mode: the base address of a 64KB page containing message based SPI signalling register aliases(0:Disabled)

Type

int

Default value

0x0

GICD-legacy-registers-as-reserved

When ARE is RAO/WI, makes superfluous registers in GICD reserved (including for the purpose of STATUSR updates)

Type

bool

Default value

0x0

GICD_CTLR.DS-1-means-secure-only

If GICD_CTLR.DS=1, GICD supports a single security state which is secure if this is true, otherwise is non-secure.

Type

bool

Default value

0x0

GICD_ITARGETSR-RAZWI

If true, the GICD_ITARGETS registers are RAZ/WI

Type

bool

Default value

0x0

GICD_PIDR

The value for the GICD_PIDR registers, if non-zero. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

GICD_TYPER2

GICD_TYPER2 value containing VID and VIL to define the width of vPEID for GICv4.1.

Type

int

Default value

0x0

GICR-clear-enable-supported

When true, this sets the value of the RO bit GICR_CTLR.CES with the value of the parameter allow-LPIEN-clear, making it visible to software

Type

bool

Default value

0x0

GICR-invalidate-registers-implemented

When true, the registers GICR_INVLPIR, GICR_INVALLR and GICR_SYNCR are implemented

Type

bool

Default value

0x0

GICR_PIDR

The value for the GICR_PIDR registers, if non-zero. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

GICR_PROPBASER-read-only

GICR_PROPBASER register is read-only.

Type

bool

Default value

0x0

GICR_PROPBASER-reset-value

Value of GICR_PROPBASER on reset.

Type

int

Default value

0x0

GITS_BASER0-entry-bytes

Number of bytes required per entry for GITS_BASER0 register.

Type

int

Default value

0x8

GITS_BASER0-indirect-RAZ

Indirect field for GITS_BASER0 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER0-type

Type field for GITS_BASER0 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER1-entry-bytes

Number of bytes required per entry for GITS_BASER1 register.

Type

int

Default value

0x8

GITS_BASER1-indirect-RAZ

Indirect field for GITS_BASER1 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER1-type

Type field for GITS_BASER1 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER2-entry-bytes

Number of bytes required per entry for GITS_BASER2 register.

Type

int

Default value

0x8

GITS_BASER2-indirect-RAZ

Indirect field for GITS_BASER2 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER2-type

Type field for GITS_BASER2 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER3-entry-bytes

Number of bytes required per entry for GITS_BASER3 register.

Type

int

Default value

0x8

GITS_BASER3-indirect-RAZ

Indirect field for GITS_BASER3 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER3-type

Type field for GITS_BASER3 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER4-entry-bytes

Number of bytes required per entry for GITS_BASER4 register.

Type

int

Default value

0x8

GITS_BASER4-indirect-RAZ

Indirect field for GITS_BASER4 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER4-type

Type field for GITS_BASER4 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER5-entry-bytes

Number of bytes required per entry for GITS_BASER5 register.

Type

int

Default value

0x8

GITS_BASER5-indirect-RAZ

Indirect field for GITS_BASER5 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER5-type

Type field for GITS_BASER5 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER6-entry-bytes

Number of bytes required per entry for GITS_BASER6 register.

Type

int

Default value

0x8

GITS_BASER6-indirect-RAZ

Indirect field for GITS_BASER6 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER6-type

Type field for GITS_BASER6 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_BASER7-entry-bytes

Number of bytes required per entry for GITS_BASER7 register.

Type

int

Default value

0x8

GITS_BASER7-indirect-RAZ

Indirect field for GITS_BASER7 register is RAZ/WI.

Type

bool

Default value

0x0

GITS_BASER7-type

Type field for GITS_BASER7 register. 0 = Unimplemented; 1 = Devices; 2 = Virtual Processors; 3 = Physical Processors; 4 = Collections

Type

int

Default value

0x0

GITS_PIDR

The value for the GITS_PIDR registers, if non-zero. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

ICFGR-PPI-mask

Mask for writes to ICFGR registers that configure PPIs

Type

int

Default value

0xaaaaaaaa

ICFGR-PPI-reset

Reset value for ICFGR registers that configure PPIs

Type

int

Default value

0x0

ICFGR-SGI-mask

Mask for writes to ICFGR registers that configure SGIs

Type

int

Default value

0x0

ICFGR-SGI-reset

Reset value for ICFGR registers that configure SGIs

Type

int

Default value

0xaaaaaaaa

ICFGR-SPI-mask

Mask for writes to ICFGR registers that configure SPIs

Type

int

Default value

0xaaaaaaaa

ICFGR-SPI-reset

Reset value for ICFGR registers that configure SPIs

Type

int

Default value

0x0

ICFGR-rsvd-bit

If ARE=0, the value of reserved bits i.e. bit 0,2,4..30 of ICFGRn for n>0

Type

bool

Default value

0x0

IGROUP-PPI-mask

Mask for writes to PPI bits in IGROUP registers

Type

int

Default value

0xffff

IGROUP-PPI-reset

Reset value for SGI bits in IGROUP registers

Type

int

Default value

0x0

IGROUP-SGI-mask

Mask for writes to SGI bits in IGROUP registers

Type

int

Default value

0xffff

IGROUP-SGI-reset

Reset value for SGI bits in IGROUP registers

Type

int

Default value

0x0

IIDR

GICD_IIDR and GICR_IIDR value

Type

int

Default value

0x0

IRI-ID-bits

Number of bits used to represent interrupts IDs in the Distributor and Redistributors, forced to 10 when none of LPIs, extended SPIs or extended PPIs is supported

Type

int

Default value

0x10

IROUTER-IRM-RAZ-WI

GICD_IROUTERn.InterruptRoutingMode is RAZ/WI

Type

bool

Default value

0x0

ITS-BASER-force-page-alignment

Force alignment of address written to a GITS_BASER register to the page size configured

Type

bool

Default value

0x1

ITS-ID-bits

Number of interrupt bits supported by ITS.

Type

int

Default value

0x10

ITS-MOVALL-update-collections

Whether MOVALL command updates the collection entires

Type

bool

Default value

0x0

ITS-TRANSLATE64R

Add an implementation specific register at 0x10008 supporting 64 bit TRANSLATER (dev[63:32], interrupt[31:0])

Type

bool

Default value

0x0

ITS-cache-invalidate-on-disable

Sets the RO bit GITS_TYPER.INV. When true, after the following sequence: 1) GITS_CTLR.Enabled written 1-->0, 2) GITS_CTLR.Quirescent observed as 1, 3) GITS_BASER<n>.Valid written 1-->0, there is no cached information from the ITS memory structure pointed to by GITS_BASER<n>.

Type

bool

Default value

0x0

ITS-collection-ID-bits

Number of collection bits supported by ITS (optional parameter, 0 => 16bits support and GITS_TYPER.CIL=0)

Type

int

Default value

0x0

ITS-count

Number of Interrupt Translation Services to be instantiated (0=none)

Type

int

Default value

0x0

ITS-cumulative-collection-tables

When true, the supported amount of collections is the sum of GITS_TYPER.HCC and the number of collections supported in memory, otherwise, simply the number supported in memory only. Irrelevant when HCC=0

Type

bool

Default value

0x1

ITS-device-bits

Number of bits supported for ITS device IDs.

Type

int

Default value

0x10

ITS-enable-itt-address-verification

If true, a transaction will be sent to ITT Address for verification.

Type

bool

Default value

0x0

ITS-entry-size

Number of bytes required to store each entry in the ITT tables.

Type

int

Default value

0x8

ITS-hardware-collection-count

Number of hardware collections held exclusively in the ITS

Type

int

Default value

0x0

ITS-legacy-iidr-typer-offset

Put the GITS_IIDR and GITS_TYPER registers at their older offset of 0x8 and 0x4 respectively

Type

bool

Default value

0x0

ITS-shared-vpe-table

Number of affinity levels to which the vPE configuration table is shared. This parameter is valid when has-gicv4.1 is true.

Type

int

Default value

0x0

ITS-threaded-command-queue

Enable execution of ITS commands in a separate thread which is sometimes required for cosimulation

Type

bool

Default value

0x1

ITS-use-physical-target-addresses

Use physical hardware addresses for targets in ITS commands -- must be true for distributed implementations

Type

bool

Default value

0x1

ITS-vmovp-bit

Device supports software issuing a VMOVPE to only one of the ITSs that has a mapping for a vPE. The device itself ensures synchronization of the VMOVPE command across all ITSs that have mapping for that vPE.

Type

bool

Default value

0x0

ITS0-base

Register base address for ITS0 (automatic if 0).

Type

int

Default value

0x0

ITS1-base

Register base address for ITS1 (automatic if 0).

Type

int

Default value

0x0

ITS2-base

Register base address for ITS2 (automatic if 0).

Type

int

Default value

0x0

ITS3-base

Register base address for ITS3 (automatic if 0).

Type

int

Default value

0x0

LPI-cache-check-data

Enable Cached LPI data against memory checking when available for cache type

Type

bool

Default value

0x0

LPI-cache-type

Cache type for LPIs, 0:No caching, 1:Full caching

Type

int

Default value

0x1

MSI_IIDR

Value returned in MSI_IIDR registers.

Type

int

Default value

0x0

MSI_NS-frame0-base

If non-zero, sets the base address used for non-secure MSI frame 0 registers.

Type

int

Default value

0x0

MSI_NS-frame0-max-SPI

Maximum SPI ID supported by non-secure MSI frame 0. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame0-min-SPI

Minimum SPI ID supported by non-secure MSI frame 0. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame1-base

If non-zero, sets the base address used for non-secure MSI frame 1 registers.

Type

int

Default value

0x0

MSI_NS-frame1-max-SPI

Maximum SPI ID supported by non-secure MSI frame 1. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame1-min-SPI

Minimum SPI ID supported by non-secure MSI frame 1. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame2-base

If non-zero, sets the base address used for non-secure MSI frame 2 registers.

Type

int

Default value

0x0

MSI_NS-frame2-max-SPI

Maximum SPI ID supported by non-secure MSI frame 2. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame2-min-SPI

Minimum SPI ID supported by non-secure MSI frame 2. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame3-base

If non-zero, sets the base address used for non-secure MSI frame 3 registers.

Type

int

Default value

0x0

MSI_NS-frame3-max-SPI

Maximum SPI ID supported by non-secure MSI frame 3. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame3-min-SPI

Minimum SPI ID supported by non-secure MSI frame 3. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame4-base

If non-zero, sets the base address used for non-secure MSI frame 4 registers.

Type

int

Default value

0x0

MSI_NS-frame4-max-SPI

Maximum SPI ID supported by non-secure MSI frame 4. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame4-min-SPI

Minimum SPI ID supported by non-secure MSI frame 4. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame5-base

If non-zero, sets the base address used for non-secure MSI frame 5 registers.

Type

int

Default value

0x0

MSI_NS-frame5-max-SPI

Maximum SPI ID supported by non-secure MSI frame 5. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame5-min-SPI

Minimum SPI ID supported by non-secure MSI frame 5. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame6-base

If non-zero, sets the base address used for non-secure MSI frame 6 registers.

Type

int

Default value

0x0

MSI_NS-frame6-max-SPI

Maximum SPI ID supported by non-secure MSI frame 6. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame6-min-SPI

Minimum SPI ID supported by non-secure MSI frame 6. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame7-base

If non-zero, sets the base address used for non-secure MSI frame 7 registers.

Type

int

Default value

0x0

MSI_NS-frame7-max-SPI

Maximum SPI ID supported by non-secure MSI frame 7. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_NS-frame7-min-SPI

Minimum SPI ID supported by non-secure MSI frame 7. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_PIDR

The value for the MSI_PIDR registers, if non-zero and distributor supports GICv2m. Note: fixed fields (device type etc.) will be overridden in this value.

Type

int

Default value

0x0

MSI_S-frame0-base

If non-zero, sets the base address used for secure MSI frame 0 registers.

Type

int

Default value

0x0

MSI_S-frame0-max-SPI

Maximum SPI ID supported by secure MSI frame 0. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame0-min-SPI

Minimum SPI ID supported by secure MSI frame 0. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame1-base

If non-zero, sets the base address used for secure MSI frame 1 registers.

Type

int

Default value

0x0

MSI_S-frame1-max-SPI

Maximum SPI ID supported by secure MSI frame 1. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame1-min-SPI

Minimum SPI ID supported by secure MSI frame 1. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame2-base

If non-zero, sets the base address used for secure MSI frame 2 registers.

Type

int

Default value

0x0

MSI_S-frame2-max-SPI

Maximum SPI ID supported by secure MSI frame 2. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame2-min-SPI

Minimum SPI ID supported by secure MSI frame 2. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame3-base

If non-zero, sets the base address used for secure MSI frame 3 registers.

Type

int

Default value

0x0

MSI_S-frame3-max-SPI

Maximum SPI ID supported by secure MSI frame 3. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame3-min-SPI

Minimum SPI ID supported by secure MSI frame 3. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame4-base

If non-zero, sets the base address used for secure MSI frame 4 registers.

Type

int

Default value

0x0

MSI_S-frame4-max-SPI

Maximum SPI ID supported by secure MSI frame 4. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame4-min-SPI

Minimum SPI ID supported by secure MSI frame 4. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame5-base

If non-zero, sets the base address used for secure MSI frame 5 registers.

Type

int

Default value

0x0

MSI_S-frame5-max-SPI

Maximum SPI ID supported by secure MSI frame 5. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame5-min-SPI

Minimum SPI ID supported by secure MSI frame 5. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame6-base

If non-zero, sets the base address used for secure MSI frame 6 registers.

Type

int

Default value

0x0

MSI_S-frame6-max-SPI

Maximum SPI ID supported by secure MSI frame 6. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame6-min-SPI

Minimum SPI ID supported by secure MSI frame 6. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame7-base

If non-zero, sets the base address used for secure MSI frame 7 registers.

Type

int

Default value

0x0

MSI_S-frame7-max-SPI

Maximum SPI ID supported by secure MSI frame 7. Set to 0 to disable frame.

Type

int

Default value

0x0

MSI_S-frame7-min-SPI

Minimum SPI ID supported by secure MSI frame 7. Set to 0 to disable frame.

Type

int

Default value

0x0

PA_SIZE

Number of valid bits in physical address

Type

int

Default value

0x30

PPI-implemented-mask

Mask of PPIs that are implemented. One bit per PPI bit 0 == PPI 16 (first PPI). This will affect other masks.

Type

int

Default value

0xffff

SPI-count

Number of SPIs that are implemented.

Type

int

Default value

0xe0

SPI-message-based-support

Distributor supports message based signaling of SPI

Type

bool

Default value

0x1

SPI-unimplemented

A comma separated list of unimplemented SPIs ranges for sparse SPI definition(for ex: '35, 39-42, 73')

Type

string

Default value**STATUSR-implemented**

Determines whether the GICR_STATUSR register is implemented.

Type

bool

Default value

0x1

add-output-cpu-wake-request-signal-from-redistributor

if true, the redistributor will have the output signal cpu_wake_request from GIC to DSU and if false, the signals are not added to the redistributor

Type

bool

Default value

0x0

allow-LPIEN-clear

Allow RW behaviour on GICR_CTLR.LPIEN instead of set once

Type

bool

Default value

0x0

clear-ISPENDR-bit-for-level-sensitive-interrupt-when-acknowledged

If true, acknowledgement of a level sensitive interrupt clears the corresponding bit in the ISPENDR register.

Type

bool

Default value

0x0

common-lpi-configuration

Describes which re-distributors share (and must be configured with the same) LPI configuration table as described in GICR_TYPER(0:All, 1:A.x.x.x, 2:A.B.x.x, 3:A.B.C.x

Type

int

Default value

0x0

common-vPE-table-affinity

Affinity value list in the form of 'a.b.c.d, e.f.g.h, etc' given to the ITS(s) (where a.b.c.d corresponds to ITS0, e.f.g.h corresponds to ITS1 and so on). Under an affinity value the vPE configuration table is shared among redistributors where the level to be shared is defined by ITS-shared-vPE-table. This parameter is valid when has-gicv4.1 is true.

Type

string

Default value**consolidators**

Specify consolidators' base addresses, interrupt line counts and base interrupt IDs, in the form 'baseAddr0:itlineCount0:baseINTID0, baseAddr1:itlineCount1:baseINTID1, [etc]' (eg '0x3f100000:64:4096, 0x3f200000:64:4224'). The consolidators' count is inferred from the list (maximum of 4). If not specified, the component contains no consolidators.

Type

string

Default value**delay-ITS-accesses**

Delay accesses from the ITS until GICR_SYNCRR is read.

Type

bool

Default value

0x1

delay-redistributor-accesses

Delay memory accesses from the redistributor until GICR_SYNCRR is read.

Type

bool

Default value

0x1

direct-lpi-support

Enable support for LPI operations through GICR registers

Type

bool

Default value

0x0

enable_protocol_checking

Enable/disable protocol checking at cpu interface

Type

bool

Default value

0x0

enabled

Enable GICv3 functionality; when false the component is inactive.

Type

bool

Default value

0x1

extended-ppi-count

Number of extended PPI supported

Type

int

Default value

0x0

extended-spi-count

Number of extended SPI supported

Type

int

Default value

0x0

fixed-routed-spis

Value of IROUTER[n] register in the form 'n=a.b.c.d, n=*'. The RM bit of IROUTER is 0 when n=a.b.c.d is used else 1 when n=* is used. n can be >= 32 and <= 1019

Type

string

Default value

gicr-icfgr-extended-count

Number of extended GICR_ICFGR registers supported

Type

int

Default value

0x4

gicv2-only

If true, when using the GICv3 model, pretend to be a GICv2 system

Type

bool

Default value

0x0

group-enables-control-doorbell

When true, GICR_VPENDBASER.{VGrp0En,VGrp1En} are cached to allow GIC to check group enables when virtual interrupt targeting this VCPU which is non-resident reaches Redistributor.

Type

bool

Default value

0x0

has-gicv4.1

Enable GICv4.1 functionality; when false the component is inactive.

Type

bool

Default value

0x0

has-two-security-states

If true, has two security states

Type

bool

Default value

0x1

has_VPENDBASER-dirty-flag-on-load

GICR_VPENDBASER.Dirty reflects transient loading state when valid=1

Type

bool

Default value

0x0

has_mpam

Enable MPAM support on ITS and RDs

Type

bool

Default value

0x0

has_nmi

Enable support for Non-maskable Interrupts (NMIs). (FEAT_GICv3_NMI)

Type

bool

Default value

0x0

ignore-generate-sgi-when-no-are

Ignore GenerateSGI packets coming from the CPU interface if both ARE_S and ARE_NS are 0

Type

bool

Default value

0x0

individual-doorbell-not-supported

For IRI with support of virtual interrupt, individual doorbell is not supported when true.

Type

bool

Default value

0x0

irouter-default-mask

Default Mask value for IROUTER[32..1019] register in the form 'a.b.c.d'

Type

string

Default value**irouter-default-reset**

Default Reset Value of IROUTER[32..1019] register in the form 'a.b.c.d' or *

Type

string

Default value**irouter-mask-values**

Mask Value of IROUTER[n] register in the form 'n=a.b.c.d'.n can be ≥ 32 and ≤ 1019

Type

string

Default value**irouter-reset-values**

Reset Value of IROUTER[n] register in the form 'n=a.b.c.d or n=*'.n can be ≥ 32 and ≤ 1019

Type

string

Default value**legacy-sgi-enable-rao**

Enables for SGI associated with an ARE=0 regime are RAO/WI

Type

bool

Default value

0x0

local-SEIs

Generate SEI to signal internal issues

Type

bool

Default value

0x0

local-VSEIs

Generate VSEI to signal internal issues

Type

bool

Default value

0x0

lockable-SPI-count

Number of SPIs that are locked down when CFGSDISABLE signal is asserted. Only applies for GICv2.

Type

int

Default value

0x0

monolithic

Indicate that the implementation is not distributed

Type

bool

Default value

0x0

mpam_max_partid

Maximum valid PARTID

Type

int

Default value

0xffff

mpam_max_pmg

Maximum valid PMG

Type

int

Default value

0xff

non-ARE-core-count

Maximum number of non-ARE cores; normally used to pass the cluster-level NUM_CORES parameter to the top-level redistributor.

Type

int

Default value

0x8

outer-cacheability-support

Allow configuration of outer cachability attributes in ITS and Redistributor

Type

bool

Default value

0x0

output_attributes

User-defined transform to be applied to bus attributes like MasterID, ExtendedID or UserFlags. Currently, only works for MPAM Attributes encoding into bus attributes.

Type

string

Default valueExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID,
ExtendedID[38]=MPAM_NS**print-memory-map**

Print memory map to stdout

Type

bool

Default value

0x0

priority-bits

Number of implemented priority bits

Type

int

Default value

0x5

processor-numbersSpecify processor numbers (as appears in GICR_TYPER) in the form 0.0.0.0=0,0.0.0.1=1 etc.)
If not specified, will number processors starting at 0.**Type**

string

Default value**redistributor-threaded-sync**Enable execution of redistributor delayed transactions in a separate thread which is
sometimes required for cosimulation**Type**

bool

Default value

0x1

reg-base

Base for decoding GICv3 registers.

Type

int

Default value

0x2c010000

reg-base-per-redistributor

Base address for each redistributor in the form '0.0.0.0=0x2c010000, 0.0.0.1=0x2c020000'. All redistributors must be specified and this overrides the reg-base parameter (except that reg-base will still be used for the top-level redistributor). If reg-base-per-redistributor-file is specified, this parameter is ignored.

Type

string

Default value**reg-base-per-redistributor-file**

Path to file containing the base address for each redistributor in the form '0.0.0.0=0x2c010000, 0.0.0.1=0x2c020000'. All redistributors must be specified and this overrides the reg-base parameter (except that reg-base will still be used for the top-level redistributor). If this parameter is specified, reg-base-per-redistributor parameter will be ignored even when it is given.

Type

string

Default value**report-MSI-error-via-statusr**

Report MSI error via GITS_STATUSR. (0:unsupported, 1:report by GITS_STATUSR, 2:report by GITS_STATUSR and interrupt as well).

Type

int

Default value

0x0

rme_default_mecid_nonsecure

Default MECID value for NON-SECURE PAS

Type

int

Default value

0x0

sgi-range-selector-support

Device has support for the Range Selector feature for SGI

Type

bool

Default value

0x0

single-set-support

When true, forces redistributors to recall interrupts with a clear rather than issue a second Set command

Type

bool

Default value

0x0

supports-shareability

Device supports shareability attributes on outgoing memory bus (i.e. is modelling an ACElite port rather than an AXI4 port).

Type

bool

Default value

0x1

trace-speculative-lpi-property-update

Trace LPI property updates performed on speculative accesses (useful for debugging LPI)

Type

bool

Default value

0x0

vPE-table-entry-size-in-doubleword

The size of one entry of a vPE configuration table in double word. The value decremented by one is shown at GICR_VPROPBASER.Entry_Size. Current model mandates the minimum entry size to be 4 doublewords. When lower value is given, it is truncated to 4.

Type

int

Default value

0x5

virtual-lpi-support

GICv4 Virtual LPIs and Direct injection of Virtual LPIs supported

Type

bool

Default value

0x0

virtual-priority-bits

Number of implemented virtual priority bits

Type

int

Default value

0x5

wakeup-on-reset

Go against specification and start redistributors in woken-up state at reset. This allows software that was written for previous versions of the GICv3 specification to work correctly. This should not be used for production code or when the distributor is used separately from the core fast model.

Type

bool

Default value

0x0

3.10.44 ICS307

Serially Programmable Clock Source. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1005: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for ICS307

This model has the following Iris instances:

Table 3-1006: ICS307 Iris instances

InstanceName	ComponentName
ICS307	ICS307
ICS307.clkdiv_clk1	ClockDivider

This model has the following MTI trace components:

Table 3-1007: ICS307 MTI instances

InstanceName	ComponentName
ICS307.clkdiv_clk1	ClockDivider

ICS307 contains the following CADI targets:

- [ClockDivider](#)

- ICS307

About ICS307

Use this component to convert the rate of one ClockSignal to another ClockSignal by using configurable multiplier, divider, and scale values. The divider ratio can be set by startup parameters or at runtime by a configuration port. Changes to the input ClockSignal rate and divider ratio are reflected immediately by the output ClockSignal ports.

Three values determine the divisor ratio:

- `vdw`.
- `rdw`.
- `od`.

To calculate the divisor ratio, use:

$$\text{Divisor} = ((\text{rdw}+2) * \text{scale}) / (2 * (\text{vdw}+8))$$

where `scale` is derived from this table indexed by `od`:

Table 3-1008: od to scale conversion

od	scale
0	10
1	2
2	8
3	4
4	5
5	7
6	3
7	6

The default values of `vdw`, `rdw` and `od` are 4, 6, and 3 to give a default divisor rate of:

$$((6+2) * 4) / (2 * (4+8)) = 4/3$$

Ports for ICS307

Table 3-1009: Ports

Name	Protocol	Type	Description
<code>clk_in</code>	ClockSignal	Slave	Master clock rate.
<code>clk_out_clk1</code>	ClockSignal	Master	Modified clock rate.
<code>clk_out_ref</code>	ClockSignal	Master	Pass through of master clock rate for divider chaining.
<code>configuration</code>	ICS307Configuration	Slave	Configuration port for setting divider ratio dynamically.

Parameters for ICS307

clkdiv_clk1.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv_clk1.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

od

OD

Type

int

Default value

0x3

rdr

RDR

Type

int

Default value

0x6

vdw

VDW

Type

int

Default value

0x4

3.10.45 IDAU

IDAU is device which provides Security attribute relating to the address pass to it. For each memory access (data and instruction), the CPU checks the IDAU and sets the security of its transactions based on it. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1010: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for IDAU

This model has the following Iris instances:

Table 3-1011: IDAU Iris instances

InstanceName	ComponentName
IDAU.bus_bridge	PVBusBridge

About IDAU

An Implementation Defined Attribution Unit (IDAU) is a device that provides a security attribute relating to an address passed to it. For each memory access (data and instructions), the CPU checks the IDAU and sets the security of its transactions based on it.

The IDAU model uses the `pv::IDAUSignal` struct to return the security attributes for the address that is passed to it. Unlike the hardware, the CPU Fast Model does not query the IDAU for each access. Communication is at a higher abstraction level to maintain simulation speed.

The fields of the `pv::IDAUSignal` struct map to the hardware signals as follows:

Table 3-1012: Mappings between `pv::IDAUSignal` fields and hardware signals

Hardware signal	<code>pv::IDAUSignal</code> field	Description
IDAUNS	<code>bool ns</code>	Non-secure region response.
IDAUNSC	<code>bool nsc</code>	Non-secure-callable region response.
IDAUID	<code>uint8_t region</code>	Region number.
IDAUIDV	<code>bool valid</code>	Region number valid.
IDAUNCHK	<code>bool exempt</code>	Region exempt from attribution check.

Masters can read or write to the `pdbus_s` port as follows:

Read

Reads return the IDAU region's `pv::IdauRegion` struct (32 bytes), which contains information about the IDAU region for the requested address. `pv::IdauRegion` contains the start address, end address, `pv::IDAUSignal` and 8 bytes of padding to make it 32-byte aligned.

Write

The port only supports 32-byte write operations to pass in a `pv::IdauRegion` struct for updating an internal IDAU region.

DMI

The port adds support for DMI requests and provides a pointer to a `pv::IdauRegion` for the requested address. An invalid DMI call back occurs if the IDAU updates its regions.



To disable the IDAU, set the `NUM_IDAU_REGION` parameter to zero.

Ports for IDAU

Table 3-1013: Ports

Name	Protocol	Type	Description
invalidate_region	Value_64	Master	This port is used as a call back to inform masters that the IDAU has updated its region information.
pvbus_s	PVBus	Slave	-

Parameters for IDAU

IDAU_REGION0.BADDR

Base address of IDAU region0

Type

int

Default value

0x0

IDAU_REGION0.ENABLE

Controls if region 0 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION0.EXEMPT

Mark IDAU region0 as exempt

Type

bool

Default value

0x0

IDAU_REGION0.LADDR

Limit address of IDAU region0

Type

int

Default value

0x0

IDAU_REGION0.NSC

Controls if region 0 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION1.BADDR

Base address of IDAU region1

Type

int

Default value

0x0

IDAU_REGION1.ENABLE

Controls if region 1 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION1.EXEMPT

Mark IDAU region1 as exempt

Type

bool

Default value

0x0

IDAU_REGION1.LADDR

Limit address of IDAU region1

Type

int

Default value

0x0

IDAU_REGION1.NSC

Controls if region 1 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION10.BADDR

Base address of IDAU region10

Type

int

Default value

0x0

IDAU_REGION10.ENABLE

Controls if region 10 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION10.EXEMPT

Mark IDAU region10 as exempt

Type

bool

Default value

0x0

IDAU_REGION10.LADDR

Limit address of IDAU region10

Type

int

Default value

0x0

IDAU_REGION10.NSC

Controls if region 10 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION100.BADDR

Base address of IDAU region100

Type

int

Default value

0x0

IDAU_REGION100.ENABLE

Controls if region 100 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION100.EXEMPT

Mark IDAU region100 as exempt

Type

bool

Default value

0x0

IDAU_REGION100.LADDR

Limit address of IDAU region100

Type

int

Default value

0x0

IDAU_REGION100.NSC

Controls if region 100 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION101.BADDR

Base address of IDAU region101

Type

int

Default value

0x0

IDAU_REGION101.ENABLE

Controls if region 101 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION101.EXEMPT

Mark IDAU region101 as exempt

Type

bool

Default value

0x0

IDAU_REGION101.LADDR

Limit address of IDAU region101

Type

int

Default value

0x0

IDAU_REGION101.NSC

Controls if region 101 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION102.BADDR

Base address of IDAU region102

Type

int

Default value

0x0

IDAU_REGION102.ENABLE

Controls if region 102 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION102.EXEMPT

Mark IDAU region102 as exempt

Type

bool

Default value

0x0

IDAU_REGION102.LADDR

Limit address of IDAU region102

Type

int

Default value

0x0

IDAU_REGION102.NSC

Controls if region 102 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION103.BADDR

Base address of IDAU region103

Type

int

Default value

0x0

IDAU_REGION103.ENABLE

Controls if region 103 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION103.EXEMPT

Mark IDAU region103 as exempt

Type

bool

Default value

0x0

IDAU_REGION103.LADDR

Limit address of IDAU region103

Type

int

Default value

0x0

IDAU_REGION103.NSC

Controls if region 103 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION104.BADDR

Base address of IDAU region104

Type

int

Default value

0x0

IDAU_REGION104.ENABLE

Controls if region 104 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION104.EXEMPT

Mark IDAU region104 as exempt

Type

bool

Default value

0x0

IDAU_REGION104.LADDR

Limit address of IDAU region104

Type

int

Default value

0x0

IDAU_REGION104.NSC

Controls if region 104 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION105.BADDR

Base address of IDAU region105

Type

int

Default value

0x0

IDAU_REGION105.ENABLE

Controls if region 105 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION105.EXEMPT

Mark IDAU region105 as exempt

Type

bool

Default value

0x0

IDAU_REGION105.LADDR

Limit address of IDAU region105

Type

int

Default value

0x0

IDAU_REGION105.NSC

Controls if region 105 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION106.BADDR

Base address of IDAU region106

Type

int

Default value

0x0

IDAU_REGION106.ENABLE

Controls if region 106 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION106.EXEMPT

Mark IDAU region106 as exempt

Type

bool

Default value

0x0

IDAU_REGION106.LADDR

Limit address of IDAU region106

Type

int

Default value

0x0

IDAU_REGION106.NSC

Controls if region 106 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION107.BADDR

Base address of IDAU region107

Type

int

Default value

0x0

IDAU_REGION107.ENABLE

Controls if region 107 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION107.EXEMPT

Mark IDAU region107 as exempt

Type

bool

Default value

0x0

IDAU_REGION107.LADDR

Limit address of IDAU region107

Type

int

Default value

0x0

IDAU_REGION107.NSC

Controls if region 107 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION108.BADDR

Base address of IDAU region108

Type

int

Default value

0x0

IDAU_REGION108.ENABLE

Controls if region 108 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION108.EXEMPT

Mark IDAU region108 as exempt

Type

bool

Default value

0x0

IDAU_REGION108.LADDR

Limit address of IDAU region108

Type

int

Default value

0x0

IDAU_REGION108.NSC

Controls if region 108 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION109.BADDR

Base address of IDAU region109

Type

int

Default value

0x0

IDAU_REGION109.ENABLE

Controls if region 109 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION109.EXEMPT

Mark IDAU region109 as exempt

Type

bool

Default value

0x0

IDAU_REGION109.LADDR

Limit address of IDAU region109

Type

int

Default value

0x0

IDAU_REGION109.NSC

Controls if region 109 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION11.BADDR

Base address of IDAU region11

Type

int

Default value

0x0

IDAU_REGION11.ENABLE

Controls if region 11 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION11.EXEMPT

Mark IDAU region11 as exempt

Type

bool

Default value

0x0

IDAU_REGION11.LADDR

Limit address of IDAU region11

Type

int

Default value

0x0

IDAU_REGION11.NSC

Controls if region 11 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION110.BADDR

Base address of IDAU region110

Type

int

Default value

0x0

IDAU_REGION110.ENABLE

Controls if region 110 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION110.EXEMPT

Mark IDAU region110 as exempt

Type

bool

Default value

0x0

IDAU_REGION110.LADDR

Limit address of IDAU region110

Type

int

Default value

0x0

IDAU_REGION110.NSC

Controls if region 110 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION111.BADDR

Base address of IDAU region111

Type

int

Default value

0x0

IDAU_REGION111.ENABLE

Controls if region 111 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION111.EXEMPT

Mark IDAU region111 as exempt

Type

bool

Default value

0x0

IDAU_REGION111.LADDR

Limit address of IDAU region111

Type

int

Default value

0x0

IDAU_REGION111.NSC

Controls if region 111 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION112.BADDR

Base address of IDAU region112

Type

int

Default value

0x0

IDAU_REGION112.ENABLE

Controls if region 112 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION112.EXEMPT

Mark IDAU region112 as exempt

Type

bool

Default value

0x0

IDAU_REGION112.LADDR

Limit address of IDAU region112

Type

int

Default value

0x0

IDAU_REGION112.NSC

Controls if region 112 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION113.BADDR

Base address of IDAU region113

Type

int

Default value

0x0

IDAU_REGION113.ENABLE

Controls if region 113 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION113.EXEMPT

Mark IDAU region113 as exempt

Type

bool

Default value

0x0

IDAU_REGION113.LADDR

Limit address of IDAU region113

Type

int

Default value

0x0

IDAU_REGION113.NSC

Controls if region 113 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION114.BADDR

Base address of IDAU region114

Type

int

Default value

0x0

IDAU_REGION114.ENABLE

Controls if region 114 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION114.EXEMPT

Mark IDAU region114 as exempt

Type

bool

Default value

0x0

IDAU_REGION114.LADDR

Limit address of IDAU region114

Type

int

Default value

0x0

IDAU_REGION114.NSC

Controls if region 114 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION115.BADDR

Base address of IDAU region115

Type

int

Default value

0x0

IDAU_REGION115.ENABLE

Controls if region 115 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION115.EXEMPT

Mark IDAU region115 as exempt

Type

bool

Default value

0x0

IDAU_REGION115.LADDR

Limit address of IDAU region115

Type

int

Default value

0x0

IDAU_REGION115.NSC

Controls if region 115 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION116.BADDR

Base address of IDAU region116

Type

int

Default value

0x0

IDAU_REGION116.ENABLE

Controls if region 116 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION116.EXEMPT

Mark IDAU region116 as exempt

Type

bool

Default value

0x0

IDAU_REGION116.LADDR

Limit address of IDAU region116

Type

int

Default value

0x0

IDAU_REGION116.NSC

Controls if region 116 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION117.BADDR

Base address of IDAU region117

Type

int

Default value

0x0

IDAU_REGION117.ENABLE

Controls if region 117 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION117.EXEMPT

Mark IDAU region117 as exempt

Type

bool

Default value

0x0

IDAU_REGION117.LADDR

Limit address of IDAU region117

Type

int

Default value

0x0

IDAU_REGION117.NSC

Controls if region 117 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION118.BADDR

Base address of IDAU region118

Type

int

Default value

0x0

IDAU_REGION118.ENABLE

Controls if region 118 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION118.EXEMPT

Mark IDAU region118 as exempt

Type

bool

Default value

0x0

IDAU_REGION118.LADDR

Limit address of IDAU region118

Type

int

Default value

0x0

IDAU_REGION118.NSC

Controls if region 118 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION119.BADDR

Base address of IDAU region119

Type

int

Default value

0x0

IDAU_REGION119.ENABLE

Controls if region 119 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION119.EXEMPT

Mark IDAU region119 as exempt

Type

bool

Default value

0x0

IDAU_REGION119.LADDR

Limit address of IDAU region119

Type

int

Default value

0x0

IDAU_REGION119.NSC

Controls if region 119 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION12.BADDR

Base address of IDAU region12

Type

int

Default value

0x0

IDAU_REGION12.ENABLE

Controls if region 12 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION12.EXEMPT

Mark IDAU region12 as exempt

Type

bool

Default value

0x0

IDAU_REGION12.LADDR

Limit address of IDAU region12

Type

int

Default value

0x0

IDAU_REGION12.NSC

Controls if region 12 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION120.BADDR

Base address of IDAU region120

Type

int

Default value

0x0

IDAU_REGION120.ENABLE

Controls if region 120 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION120.EXEMPT

Mark IDAU region120 as exempt

Type

bool

Default value

0x0

IDAU_REGION120.LADDR

Limit address of IDAU region120

Type

int

Default value

0x0

IDAU_REGION120.NSC

Controls if region 120 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION121.BADDR

Base address of IDAU region121

Type

int

Default value

0x0

IDAU_REGION121.ENABLE

Controls if region 121 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION121.EXEMPT

Mark IDAU region121 as exempt

Type

bool

Default value

0x0

IDAU_REGION121.LADDR

Limit address of IDAU region121

Type

int

Default value

0x0

IDAU_REGION121.NSC

Controls if region 121 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION122.BADDR

Base address of IDAU region122

Type

int

Default value

0x0

IDAU_REGION122.ENABLE

Controls if region 122 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION122.EXEMPT

Mark IDAU region122 as exempt

Type

bool

Default value

0x0

IDAU_REGION122.LADDR

Limit address of IDAU region122

Type

int

Default value

0x0

IDAU_REGION122.NSC

Controls if region 122 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION123.BADDR

Base address of IDAU region123

Type

int

Default value

0x0

IDAU_REGION123.ENABLE

Controls if region 123 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION123.EXEMPT

Mark IDAU region123 as exempt

Type

bool

Default value

0x0

IDAU_REGION123.LADDR

Limit address of IDAU region123

Type

int

Default value

0x0

IDAU_REGION123.NSC

Controls if region 123 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION124.BADDR

Base address of IDAU region124

Type

int

Default value

0x0

IDAU_REGION124.ENABLE

Controls if region 124 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION124.EXEMPT

Mark IDAU region124 as exempt

Type

bool

Default value

0x0

IDAU_REGION124.LADDR

Limit address of IDAU region124

Type

int

Default value

0x0

IDAU_REGION124.NSC

Controls if region 124 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION125.BADDR

Base address of IDAU region125

Type

int

Default value

0x0

IDAU_REGION125.ENABLE

Controls if region 125 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION125.EXEMPT

Mark IDAU region125 as exempt

Type

bool

Default value

0x0

IDAU_REGION125.LADDR

Limit address of IDAU region125

Type

int

Default value

0x0

IDAU_REGION125.NSC

Controls if region 125 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION126.BADDR

Base address of IDAU region126

Type

int

Default value

0x0

IDAU_REGION126.ENABLE

Controls if region 126 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION126.EXEMPT

Mark IDAU region126 as exempt

Type

bool

Default value

0x0

IDAU_REGION126.LADDR

Limit address of IDAU region126

Type

int

Default value

0x0

IDAU_REGION126.NSC

Controls if region 126 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION127.BADDR

Base address of IDAU region127

Type

int

Default value

0x0

IDAU_REGION127.ENABLE

Controls if region 127 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION127.EXEMPT

Mark IDAU region127 as exempt

Type

bool

Default value

0x0

IDAU_REGION127.LADDR

Limit address of IDAU region127

Type

int

Default value

0x0

IDAU_REGION127.NSC

Controls if region 127 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION128.BADDR

Base address of IDAU region128

Type

int

Default value

0x0

IDAU_REGION128.ENABLE

Controls if region 128 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION128.EXEMPT

Mark IDAU region128 as exempt

Type

bool

Default value

0x0

IDAU_REGION128.LADDR

Limit address of IDAU region128

Type

int

Default value

0x0

IDAU_REGION128.NSC

Controls if region 128 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION129.BADDR

Base address of IDAU region129

Type

int

Default value

0x0

IDAU_REGION129.ENABLE

Controls if region 129 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION129.EXEMPT

Mark IDAU region129 as exempt

Type

bool

Default value

0x0

IDAU_REGION129.LADDR

Limit address of IDAU region129

Type

int

Default value

0x0

IDAU_REGION129.NSC

Controls if region 129 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION13.BADDR

Base address of IDAU region13

Type

int

Default value

0x0

IDAU_REGION13.ENABLE

Controls if region 13 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION13.EXEMPT

Mark IDAU region13 as exempt

Type

bool

Default value

0x0

IDAU_REGION13.LADDR

Limit address of IDAU region13

Type

int

Default value

0x0

IDAU_REGION13.NSC

Controls if region 13 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION130.BADDR

Base address of IDAU region130

Type

int

Default value

0x0

IDAU_REGION130.ENABLE

Controls if region 130 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION130.EXEMPT

Mark IDAU region130 as exempt

Type

bool

Default value

0x0

IDAU_REGION130.LADDR

Limit address of IDAU region130

Type

int

Default value

0x0

IDAU_REGION130.NSC

Controls if region 130 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION131.BADDR

Base address of IDAU region131

Type

int

Default value

0x0

IDAU_REGION131.ENABLE

Controls if region 131 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION131.EXEMPT

Mark IDAU region131 as exempt

Type

bool

Default value

0x0

IDAU_REGION131.LADDR

Limit address of IDAU region131

Type

int

Default value

0x0

IDAU_REGION131.NSC

Controls if region 131 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION132.BADDR

Base address of IDAU region132

Type

int

Default value

0x0

IDAU_REGION132.ENABLE

Controls if region 132 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION132.EXEMPT

Mark IDAU region132 as exempt

Type

bool

Default value

0x0

IDAU_REGION132.LADDR

Limit address of IDAU region132

Type

int

Default value

0x0

IDAU_REGION132.NSC

Controls if region 132 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION133.BADDR

Base address of IDAU region133

Type

int

Default value

0x0

IDAU_REGION133.ENABLE

Controls if region 133 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION133.EXEMPT

Mark IDAU region133 as exempt

Type

bool

Default value

0x0

IDAU_REGION133.LADDR

Limit address of IDAU region133

Type

int

Default value

0x0

IDAU_REGION133.NSC

Controls if region 133 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION134.BADDR

Base address of IDAU region134

Type

int

Default value

0x0

IDAU_REGION134.ENABLE

Controls if region 134 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION134.EXEMPT

Mark IDAU region134 as exempt

Type

bool

Default value

0x0

IDAU_REGION134.LADDR

Limit address of IDAU region134

Type

int

Default value

0x0

IDAU_REGION134.NSC

Controls if region 134 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION135.BADDR

Base address of IDAU region135

Type

int

Default value

0x0

IDAU_REGION135.ENABLE

Controls if region 135 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION135.EXEMPT

Mark IDAU region135 as exempt

Type

bool

Default value

0x0

IDAU_REGION135.LADDR

Limit address of IDAU region135

Type

int

Default value

0x0

IDAU_REGION135.NSC

Controls if region 135 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION136.BADDR

Base address of IDAU region136

Type

int

Default value

0x0

IDAU_REGION136.ENABLE

Controls if region 136 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION136.EXEMPT

Mark IDAU region136 as exempt

Type

bool

Default value

0x0

IDAU_REGION136.LADDR

Limit address of IDAU region136

Type

int

Default value

0x0

IDAU_REGION136.NSC

Controls if region 136 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION137.BADDR

Base address of IDAU region137

Type

int

Default value

0x0

IDAU_REGION137.ENABLE

Controls if region 137 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION137.EXEMPT

Mark IDAU region137 as exempt

Type

bool

Default value

0x0

IDAU_REGION137.LADDR

Limit address of IDAU region137

Type

int

Default value

0x0

IDAU_REGION137.NSC

Controls if region 137 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION138.BADDR

Base address of IDAU region138

Type

int

Default value

0x0

IDAU_REGION138.ENABLE

Controls if region 138 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION138.EXEMPT

Mark IDAU region138 as exempt

Type

bool

Default value

0x0

IDAU_REGION138.LADDR

Limit address of IDAU region138

Type

int

Default value

0x0

IDAU_REGION138.NSC

Controls if region 138 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION139.BADDR

Base address of IDAU region139

Type

int

Default value

0x0

IDAU_REGION139.ENABLE

Controls if region 139 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION139.EXEMPT

Mark IDAU region139 as exempt

Type

bool

Default value

0x0

IDAU_REGION139.LADDR

Limit address of IDAU region139

Type

int

Default value

0x0

IDAU_REGION139.NSC

Controls if region 139 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION14.BADDR

Base address of IDAU region14

Type

int

Default value

0x0

IDAU_REGION14.ENABLE

Controls if region 14 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION14.EXEMPT

Mark IDAU region14 as exempt

Type

bool

Default value

0x0

IDAU_REGION14.LADDR

Limit address of IDAU region14

Type

int

Default value

0x0

IDAU_REGION14.NSC

Controls if region 14 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION140.BADDR

Base address of IDAU region140

Type

int

Default value

0x0

IDAU_REGION140.ENABLE

Controls if region 140 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION140.EXEMPT

Mark IDAU region140 as exempt

Type

bool

Default value

0x0

IDAU_REGION140.LADDR

Limit address of IDAU region140

Type

int

Default value

0x0

IDAU_REGION140.NSC

Controls if region 140 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION141.BADDR

Base address of IDAU region141

Type

int

Default value

0x0

IDAU_REGION141.ENABLE

Controls if region 141 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION141.EXEMPT

Mark IDAU region141 as exempt

Type

bool

Default value

0x0

IDAU_REGION141.LADDR

Limit address of IDAU region141

Type

int

Default value

0x0

IDAU_REGION141.NSC

Controls if region 141 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION142.BADDR

Base address of IDAU region142

Type

int

Default value

0x0

IDAU_REGION142.ENABLE

Controls if region 142 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION142.EXEMPT

Mark IDAU region142 as exempt

Type

bool

Default value

0x0

IDAU_REGION142.LADDR

Limit address of IDAU region142

Type

int

Default value

0x0

IDAU_REGION142.NSC

Controls if region 142 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION143.BADDR

Base address of IDAU region143

Type

int

Default value

0x0

IDAU_REGION143.ENABLE

Controls if region 143 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION143.EXEMPT

Mark IDAU region143 as exempt

Type

bool

Default value

0x0

IDAU_REGION143.LADDR

Limit address of IDAU region143

Type

int

Default value

0x0

IDAU_REGION143.NSC

Controls if region 143 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION144.BADDR

Base address of IDAU region144

Type

int

Default value

0x0

IDAU_REGION144.ENABLE

Controls if region 144 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION144.EXEMPT

Mark IDAU region144 as exempt

Type

bool

Default value

0x0

IDAU_REGION144.LADDR

Limit address of IDAU region144

Type

int

Default value

0x0

IDAU_REGION144.NSC

Controls if region 144 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION145.BADDR

Base address of IDAU region145

Type

int

Default value

0x0

IDAU_REGION145.ENABLE

Controls if region 145 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION145.EXEMPT

Mark IDAU region145 as exempt

Type

bool

Default value

0x0

IDAU_REGION145.LADDR

Limit address of IDAU region145

Type

int

Default value

0x0

IDAU_REGION145.NSC

Controls if region 145 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION146.BADDR

Base address of IDAU region146

Type

int

Default value

0x0

IDAU_REGION146.ENABLE

Controls if region 146 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION146.EXEMPT

Mark IDAU region146 as exempt

Type

bool

Default value

0x0

IDAU_REGION146.LADDR

Limit address of IDAU region146

Type

int

Default value

0x0

IDAU_REGION146.NSC

Controls if region 146 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION147.BADDR

Base address of IDAU region147

Type

int

Default value

0x0

IDAU_REGION147.ENABLE

Controls if region 147 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION147.EXEMPT

Mark IDAU region147 as exempt

Type

bool

Default value

0x0

IDAU_REGION147.LADDR

Limit address of IDAU region147

Type

int

Default value

0x0

IDAU_REGION147.NSC

Controls if region 147 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION148.BADDR

Base address of IDAU region148

Type

int

Default value

0x0

IDAU_REGION148.ENABLE

Controls if region 148 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION148.EXEMPT

Mark IDAU region148 as exempt

Type

bool

Default value

0x0

IDAU_REGION148.LADDR

Limit address of IDAU region148

Type

int

Default value

0x0

IDAU_REGION148.NSC

Controls if region 148 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION149.BADDR

Base address of IDAU region149

Type

int

Default value

0x0

IDAU_REGION149.ENABLE

Controls if region 149 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION149.EXEMPT

Mark IDAU region149 as exempt

Type

bool

Default value

0x0

IDAU_REGION149.LADDR

Limit address of IDAU region149

Type

int

Default value

0x0

IDAU_REGION149.NSC

Controls if region 149 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION15.BADDR

Base address of IDAU region15

Type

int

Default value

0x0

IDAU_REGION15.ENABLE

Controls if region 15 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION15.EXEMPT

Mark IDAU region15 as exempt

Type

bool

Default value

0x0

IDAU_REGION15.LADDR

Limit address of IDAU region15

Type

int

Default value

0x0

IDAU_REGION15.NSC

Controls if region 15 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION150.BADDR

Base address of IDAU region150

Type

int

Default value

0x0

IDAU_REGION150.ENABLE

Controls if region 150 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION150.EXEMPT

Mark IDAU region150 as exempt

Type

bool

Default value

0x0

IDAU_REGION150.LADDR

Limit address of IDAU region150

Type

int

Default value

0x0

IDAU_REGION150.NSC

Controls if region 150 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION151.BADDR

Base address of IDAU region151

Type

int

Default value

0x0

IDAU_REGION151.ENABLE

Controls if region 151 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION151.EXEMPT

Mark IDAU region151 as exempt

Type

bool

Default value

0x0

IDAU_REGION151.LADDR

Limit address of IDAU region151

Type

int

Default value

0x0

IDAU_REGION151.NSC

Controls if region 151 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION152.BADDR

Base address of IDAU region152

Type

int

Default value

0x0

IDAU_REGION152.ENABLE

Controls if region 152 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION152.EXEMPT

Mark IDAU region152 as exempt

Type

bool

Default value

0x0

IDAU_REGION152.LADDR

Limit address of IDAU region152

Type

int

Default value

0x0

IDAU_REGION152.NSC

Controls if region 152 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION153.BADDR

Base address of IDAU region153

Type

int

Default value

0x0

IDAU_REGION153.ENABLE

Controls if region 153 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION153.EXEMPT

Mark IDAU region153 as exempt

Type

bool

Default value

0x0

IDAU_REGION153.LADDR

Limit address of IDAU region153

Type

int

Default value

0x0

IDAU_REGION153.NSC

Controls if region 153 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION154.BADDR

Base address of IDAU region154

Type

int

Default value

0x0

IDAU_REGION154.ENABLE

Controls if region 154 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION154.EXEMPT

Mark IDAU region154 as exempt

Type

bool

Default value

0x0

IDAU_REGION154.LADDR

Limit address of IDAU region154

Type

int

Default value

0x0

IDAU_REGION154.NSC

Controls if region 154 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION155.BADDR

Base address of IDAU region155

Type

int

Default value

0x0

IDAU_REGION155.ENABLE

Controls if region 155 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION155.EXEMPT

Mark IDAU region155 as exempt

Type

bool

Default value

0x0

IDAU_REGION155.LADDR

Limit address of IDAU region155

Type

int

Default value

0x0

IDAU_REGION155.NSC

Controls if region 155 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION156.BADDR

Base address of IDAU region156

Type

int

Default value

0x0

IDAU_REGION156.ENABLE

Controls if region 156 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION156.EXEMPT

Mark IDAU region156 as exempt

Type

bool

Default value

0x0

IDAU_REGION156.LADDR

Limit address of IDAU region156

Type

int

Default value

0x0

IDAU_REGION156.NSC

Controls if region 156 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION157.BADDR

Base address of IDAU region157

Type

int

Default value

0x0

IDAU_REGION157.ENABLE

Controls if region 157 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION157.EXEMPT

Mark IDAU region157 as exempt

Type

bool

Default value

0x0

IDAU_REGION157.LADDR

Limit address of IDAU region157

Type

int

Default value

0x0

IDAU_REGION157.NSC

Controls if region 157 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION158.BADDR

Base address of IDAU region158

Type

int

Default value

0x0

IDAU_REGION158.ENABLE

Controls if region 158 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION158.EXEMPT

Mark IDAU region158 as exempt

Type

bool

Default value

0x0

IDAU_REGION158.LADDR

Limit address of IDAU region158

Type

int

Default value

0x0

IDAU_REGION158.NSC

Controls if region 158 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION159.BADDR

Base address of IDAU region159

Type

int

Default value

0x0

IDAU_REGION159.ENABLE

Controls if region 159 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION159.EXEMPT

Mark IDAU region159 as exempt

Type

bool

Default value

0x0

IDAU_REGION159.LADDR

Limit address of IDAU region159

Type

int

Default value

0x0

IDAU_REGION159.NSC

Controls if region 159 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION16.BADDR

Base address of IDAU region16

Type

int

Default value

0x0

IDAU_REGION16.ENABLE

Controls if region 16 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION16.EXEMPT

Mark IDAU region16 as exempt

Type

bool

Default value

0x0

IDAU_REGION16.LADDR

Limit address of IDAU region16

Type

int

Default value

0x0

IDAU_REGION16.NSC

Controls if region 16 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION160.BADDR

Base address of IDAU region160

Type

int

Default value

0x0

IDAU_REGION160.ENABLE

Controls if region 160 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION160.EXEMPT

Mark IDAU region160 as exempt

Type

bool

Default value

0x0

IDAU_REGION160.LADDR

Limit address of IDAU region160

Type

int

Default value

0x0

IDAU_REGION160.NSC

Controls if region 160 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION161.BADDR

Base address of IDAU region161

Type

int

Default value

0x0

IDAU_REGION161.ENABLE

Controls if region 161 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION161.EXEMPT

Mark IDAU region161 as exempt

Type

bool

Default value

0x0

IDAU_REGION161.LADDR

Limit address of IDAU region161

Type

int

Default value

0x0

IDAU_REGION161.NSC

Controls if region 161 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION162.BADDR

Base address of IDAU region162

Type

int

Default value

0x0

IDAU_REGION162.ENABLE

Controls if region 162 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION162.EXEMPT

Mark IDAU region162 as exempt

Type

bool

Default value

0x0

IDAU_REGION162.LADDR

Limit address of IDAU region162

Type

int

Default value

0x0

IDAU_REGION162.NSC

Controls if region 162 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION163.BADDR

Base address of IDAU region163

Type

int

Default value

0x0

IDAU_REGION163.ENABLE

Controls if region 163 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION163.EXEMPT

Mark IDAU region163 as exempt

Type

bool

Default value

0x0

IDAU_REGION163.LADDR

Limit address of IDAU region163

Type

int

Default value

0x0

IDAU_REGION163.NSC

Controls if region 163 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION164.BADDR

Base address of IDAU region164

Type

int

Default value

0x0

IDAU_REGION164.ENABLE

Controls if region 164 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION164.EXEMPT

Mark IDAU region164 as exempt

Type

bool

Default value

0x0

IDAU_REGION164.LADDR

Limit address of IDAU region164

Type

int

Default value

0x0

IDAU_REGION164.NSC

Controls if region 164 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION165.BADDR

Base address of IDAU region165

Type

int

Default value

0x0

IDAU_REGION165.ENABLE

Controls if region 165 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION165.EXEMPT

Mark IDAU region165 as exempt

Type

bool

Default value

0x0

IDAU_REGION165.LADDR

Limit address of IDAU region165

Type

int

Default value

0x0

IDAU_REGION165.NSC

Controls if region 165 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION166.BADDR

Base address of IDAU region166

Type

int

Default value

0x0

IDAU_REGION166.ENABLE

Controls if region 166 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION166.EXEMPT

Mark IDAU region166 as exempt

Type

bool

Default value

0x0

IDAU_REGION166.LADDR

Limit address of IDAU region166

Type

int

Default value

0x0

IDAU_REGION166.NSC

Controls if region 166 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION167.BADDR

Base address of IDAU region167

Type

int

Default value

0x0

IDAU_REGION167.ENABLE

Controls if region 167 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION167.EXEMPT

Mark IDAU region167 as exempt

Type

bool

Default value

0x0

IDAU_REGION167.LADDR

Limit address of IDAU region167

Type

int

Default value

0x0

IDAU_REGION167.NSC

Controls if region 167 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION168.BADDR

Base address of IDAU region168

Type

int

Default value

0x0

IDAU_REGION168.ENABLE

Controls if region 168 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION168.EXEMPT

Mark IDAU region168 as exempt

Type

bool

Default value

0x0

IDAU_REGION168.LADDR

Limit address of IDAU region168

Type

int

Default value

0x0

IDAU_REGION168.NSC

Controls if region 168 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION169.BADDR

Base address of IDAU region169

Type

int

Default value

0x0

IDAU_REGION169.ENABLE

Controls if region 169 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION169.EXEMPT

Mark IDAU region169 as exempt

Type

bool

Default value

0x0

IDAU_REGION169.LADDR

Limit address of IDAU region169

Type

int

Default value

0x0

IDAU_REGION169.NSC

Controls if region 169 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION17.BADDR

Base address of IDAU region17

Type

int

Default value

0x0

IDAU_REGION17.ENABLE

Controls if region 17 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION17.EXEMPT

Mark IDAU region17 as exempt

Type

bool

Default value

0x0

IDAU_REGION17.LADDR

Limit address of IDAU region17

Type

int

Default value

0x0

IDAU_REGION17.NSC

Controls if region 17 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION170.BADDR

Base address of IDAU region170

Type

int

Default value

0x0

IDAU_REGION170.ENABLE

Controls if region 170 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION170.EXEMPT

Mark IDAU region170 as exempt

Type

bool

Default value

0x0

IDAU_REGION170.LADDR

Limit address of IDAU region170

Type

int

Default value

0x0

IDAU_REGION170.NSC

Controls if region 170 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION171.BADDR

Base address of IDAU region171

Type

int

Default value

0x0

IDAU_REGION171.ENABLE

Controls if region 171 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION171.EXEMPT

Mark IDAU region171 as exempt

Type

bool

Default value

0x0

IDAU_REGION171.LADDR

Limit address of IDAU region171

Type

int

Default value

0x0

IDAU_REGION171.NSC

Controls if region 171 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION172.BADDR

Base address of IDAU region172

Type

int

Default value

0x0

IDAU_REGION172.ENABLE

Controls if region 172 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION172.EXEMPT

Mark IDAU region172 as exempt

Type

bool

Default value

0x0

IDAU_REGION172.LADDR

Limit address of IDAU region172

Type

int

Default value

0x0

IDAU_REGION172.NSC

Controls if region 172 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION173.BADDR

Base address of IDAU region173

Type

int

Default value

0x0

IDAU_REGION173.ENABLE

Controls if region 173 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION173.EXEMPT

Mark IDAU region173 as exempt

Type

bool

Default value

0x0

IDAU_REGION173.LADDR

Limit address of IDAU region173

Type

int

Default value

0x0

IDAU_REGION173.NSC

Controls if region 173 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION174.BADDR

Base address of IDAU region174

Type

int

Default value

0x0

IDAU_REGION174.ENABLE

Controls if region 174 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION174.EXEMPT

Mark IDAU region174 as exempt

Type

bool

Default value

0x0

IDAU_REGION174.LADDR

Limit address of IDAU region174

Type

int

Default value

0x0

IDAU_REGION174.NSC

Controls if region 174 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION175.BADDR

Base address of IDAU region175

Type

int

Default value

0x0

IDAU_REGION175.ENABLE

Controls if region 175 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION175.EXEMPT

Mark IDAU region175 as exempt

Type

bool

Default value

0x0

IDAU_REGION175.LADDR

Limit address of IDAU region175

Type

int

Default value

0x0

IDAU_REGION175.NSC

Controls if region 175 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION176.BADDR

Base address of IDAU region176

Type

int

Default value

0x0

IDAU_REGION176.ENABLE

Controls if region 176 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION176.EXEMPT

Mark IDAU region176 as exempt

Type

bool

Default value

0x0

IDAU_REGION176.LADDR

Limit address of IDAU region176

Type

int

Default value

0x0

IDAU_REGION176.NSC

Controls if region 176 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION177.BADDR

Base address of IDAU region177

Type

int

Default value

0x0

IDAU_REGION177.ENABLE

Controls if region 177 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION177.EXEMPT

Mark IDAU region177 as exempt

Type

bool

Default value

0x0

IDAU_REGION177.LADDR

Limit address of IDAU region177

Type

int

Default value

0x0

IDAU_REGION177.NSC

Controls if region 177 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION178.BADDR

Base address of IDAU region178

Type

int

Default value

0x0

IDAU_REGION178.ENABLE

Controls if region 178 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION178.EXEMPT

Mark IDAU region178 as exempt

Type

bool

Default value

0x0

IDAU_REGION178.LADDR

Limit address of IDAU region178

Type

int

Default value

0x0

IDAU_REGION178.NSC

Controls if region 178 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION179.BADDR

Base address of IDAU region179

Type

int

Default value

0x0

IDAU_REGION179.ENABLE

Controls if region 179 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION179.EXEMPT

Mark IDAU region179 as exempt

Type

bool

Default value

0x0

IDAU_REGION179.LADDR

Limit address of IDAU region179

Type

int

Default value

0x0

IDAU_REGION179.NSC

Controls if region 179 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION18.BADDR

Base address of IDAU region18

Type

int

Default value

0x0

IDAU_REGION18.ENABLE

Controls if region 18 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION18.EXEMPT

Mark IDAU region18 as exempt

Type

bool

Default value

0x0

IDAU_REGION18.LADDR

Limit address of IDAU region18

Type

int

Default value

0x0

IDAU_REGION18.NSC

Controls if region 18 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION180.BADDR

Base address of IDAU region180

Type

int

Default value

0x0

IDAU_REGION180.ENABLE

Controls if region 180 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION180.EXEMPT

Mark IDAU region180 as exempt

Type

bool

Default value

0x0

IDAU_REGION180.LADDR

Limit address of IDAU region180

Type

int

Default value

0x0

IDAU_REGION180.NSC

Controls if region 180 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION181.BADDR

Base address of IDAU region181

Type

int

Default value

0x0

IDAU_REGION181.ENABLE

Controls if region 181 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION181.EXEMPT

Mark IDAU region181 as exempt

Type

bool

Default value

0x0

IDAU_REGION181.LADDR

Limit address of IDAU region181

Type

int

Default value

0x0

IDAU_REGION181.NSC

Controls if region 181 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION182.BADDR

Base address of IDAU region182

Type

int

Default value

0x0

IDAU_REGION182.ENABLE

Controls if region 182 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION182.EXEMPT

Mark IDAU region182 as exempt

Type

bool

Default value

0x0

IDAU_REGION182.LADDR

Limit address of IDAU region182

Type

int

Default value

0x0

IDAU_REGION182.NSC

Controls if region 182 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION183.BADDR

Base address of IDAU region183

Type

int

Default value

0x0

IDAU_REGION183.ENABLE

Controls if region 183 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION183.EXEMPT

Mark IDAU region183 as exempt

Type

bool

Default value

0x0

IDAU_REGION183.LADDR

Limit address of IDAU region183

Type

int

Default value

0x0

IDAU_REGION183.NSC

Controls if region 183 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION184.BADDR

Base address of IDAU region184

Type

int

Default value

0x0

IDAU_REGION184.ENABLE

Controls if region 184 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION184.EXEMPT

Mark IDAU region184 as exempt

Type

bool

Default value

0x0

IDAU_REGION184.LADDR

Limit address of IDAU region184

Type

int

Default value

0x0

IDAU_REGION184.NSC

Controls if region 184 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION185.BADDR

Base address of IDAU region185

Type

int

Default value

0x0

IDAU_REGION185.ENABLE

Controls if region 185 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION185.EXEMPT

Mark IDAU region185 as exempt

Type

bool

Default value

0x0

IDAU_REGION185.LADDR

Limit address of IDAU region185

Type

int

Default value

0x0

IDAU_REGION185.NSC

Controls if region 185 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION186.BADDR

Base address of IDAU region186

Type

int

Default value

0x0

IDAU_REGION186.ENABLE

Controls if region 186 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION186.EXEMPT

Mark IDAU region186 as exempt

Type

bool

Default value

0x0

IDAU_REGION186.LADDR

Limit address of IDAU region186

Type

int

Default value

0x0

IDAU_REGION186.NSC

Controls if region 186 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION187.BADDR

Base address of IDAU region187

Type

int

Default value

0x0

IDAU_REGION187.ENABLE

Controls if region 187 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION187.EXEMPT

Mark IDAU region187 as exempt

Type

bool

Default value

0x0

IDAU_REGION187.LADDR

Limit address of IDAU region187

Type

int

Default value

0x0

IDAU_REGION187.NSC

Controls if region 187 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION188.BADDR

Base address of IDAU region188

Type

int

Default value

0x0

IDAU_REGION188.ENABLE

Controls if region 188 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION188.EXEMPT

Mark IDAU region188 as exempt

Type

bool

Default value

0x0

IDAU_REGION188.LADDR

Limit address of IDAU region188

Type

int

Default value

0x0

IDAU_REGION188.NSC

Controls if region 188 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION189.BADDR

Base address of IDAU region189

Type

int

Default value

0x0

IDAU_REGION189.ENABLE

Controls if region 189 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION189.EXEMPT

Mark IDAU region189 as exempt

Type

bool

Default value

0x0

IDAU_REGION189.LADDR

Limit address of IDAU region189

Type

int

Default value

0x0

IDAU_REGION189.NSC

Controls if region 189 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION19.BADDR

Base address of IDAU region19

Type

int

Default value

0x0

IDAU_REGION19.ENABLE

Controls if region 19 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION19.EXEMPT

Mark IDAU region19 as exempt

Type

bool

Default value

0x0

IDAU_REGION19.LADDR

Limit address of IDAU region19

Type

int

Default value

0x0

IDAU_REGION19.NSC

Controls if region 19 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION190.BADDR

Base address of IDAU region190

Type

int

Default value

0x0

IDAU_REGION190.ENABLE

Controls if region 190 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION190.EXEMPT

Mark IDAU region190 as exempt

Type

bool

Default value

0x0

IDAU_REGION190.LADDR

Limit address of IDAU region190

Type

int

Default value

0x0

IDAU_REGION190.NSC

Controls if region 190 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION191.BADDR

Base address of IDAU region191

Type

int

Default value

0x0

IDAU_REGION191.ENABLE

Controls if region 191 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION191.EXEMPT

Mark IDAU region191 as exempt

Type

bool

Default value

0x0

IDAU_REGION191.LADDR

Limit address of IDAU region191

Type

int

Default value

0x0

IDAU_REGION191.NSC

Controls if region 191 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION192.BADDR

Base address of IDAU region192

Type

int

Default value

0x0

IDAU_REGION192.ENABLE

Controls if region 192 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION192.EXEMPT

Mark IDAU region192 as exempt

Type

bool

Default value

0x0

IDAU_REGION192.LADDR

Limit address of IDAU region192

Type

int

Default value

0x0

IDAU_REGION192.NSC

Controls if region 192 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION193.BADDR

Base address of IDAU region193

Type

int

Default value

0x0

IDAU_REGION193.ENABLE

Controls if region 193 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION193.EXEMPT

Mark IDAU region193 as exempt

Type

bool

Default value

0x0

IDAU_REGION193.LADDR

Limit address of IDAU region193

Type

int

Default value

0x0

IDAU_REGION193.NSC

Controls if region 193 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION194.BADDR

Base address of IDAU region194

Type

int

Default value

0x0

IDAU_REGION194.ENABLE

Controls if region 194 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION194.EXEMPT

Mark IDAU region194 as exempt

Type

bool

Default value

0x0

IDAU_REGION194.LADDR

Limit address of IDAU region194

Type

int

Default value

0x0

IDAU_REGION194.NSC

Controls if region 194 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION195.BADDR

Base address of IDAU region195

Type

int

Default value

0x0

IDAU_REGION195.ENABLE

Controls if region 195 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION195.EXEMPT

Mark IDAU region195 as exempt

Type

bool

Default value

0x0

IDAU_REGION195.LADDR

Limit address of IDAU region195

Type

int

Default value

0x0

IDAU_REGION195.NSC

Controls if region 195 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION196.BADDR

Base address of IDAU region196

Type

int

Default value

0x0

IDAU_REGION196.ENABLE

Controls if region 196 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION196.EXEMPT

Mark IDAU region196 as exempt

Type

bool

Default value

0x0

IDAU_REGION196.LADDR

Limit address of IDAU region196

Type

int

Default value

0x0

IDAU_REGION196.NSC

Controls if region 196 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION197.BADDR

Base address of IDAU region197

Type

int

Default value

0x0

IDAU_REGION197.ENABLE

Controls if region 197 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION197.EXEMPT

Mark IDAU region197 as exempt

Type

bool

Default value

0x0

IDAU_REGION197.LADDR

Limit address of IDAU region197

Type

int

Default value

0x0

IDAU_REGION197.NSC

Controls if region 197 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION198.BADDR

Base address of IDAU region198

Type

int

Default value

0x0

IDAU_REGION198.ENABLE

Controls if region 198 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION198.EXEMPT

Mark IDAU region198 as exempt

Type

bool

Default value

0x0

IDAU_REGION198.LADDR

Limit address of IDAU region198

Type

int

Default value

0x0

IDAU_REGION198.NSC

Controls if region 198 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION199.BADDR

Base address of IDAU region199

Type

int

Default value

0x0

IDAU_REGION199.ENABLE

Controls if region 199 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION199.EXEMPT

Mark IDAU region199 as exempt

Type

bool

Default value

0x0

IDAU_REGION199.LADDR

Limit address of IDAU region199

Type

int

Default value

0x0

IDAU_REGION199.NSC

Controls if region 199 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION2.BADDR

Base address of IDAU region2

Type

int

Default value

0x0

IDAU_REGION2.ENABLE

Controls if region 2 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION2.EXEMPT

Mark IDAU region2 as exempt

Type

bool

Default value

0x0

IDAU_REGION2.LADDR

Limit address of IDAU region2

Type

int

Default value

0x0

IDAU_REGION2.NSC

Controls if region 2 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION20.BADDR

Base address of IDAU region20

Type

int

Default value

0x0

IDAU_REGION20.ENABLE

Controls if region 20 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION20.EXEMPT

Mark IDAU region20 as exempt

Type

bool

Default value

0x0

IDAU_REGION20.LADDR

Limit address of IDAU region20

Type

int

Default value

0x0

IDAU_REGION20.NSC

Controls if region 20 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION200.BADDR

Base address of IDAU region200

Type

int

Default value

0x0

IDAU_REGION200.ENABLE

Controls if region 200 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION200.EXEMPT

Mark IDAU region200 as exempt

Type

bool

Default value

0x0

IDAU_REGION200.LADDR

Limit address of IDAU region200

Type

int

Default value

0x0

IDAU_REGION200.NSC

Controls if region 200 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION201.BADDR

Base address of IDAU region201

Type

int

Default value

0x0

IDAU_REGION201.ENABLE

Controls if region 201 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION201.EXEMPT

Mark IDAU region201 as exempt

Type

bool

Default value

0x0

IDAU_REGION201.LADDR

Limit address of IDAU region201

Type

int

Default value

0x0

IDAU_REGION201.NSC

Controls if region 201 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION202.BADDR

Base address of IDAU region202

Type

int

Default value

0x0

IDAU_REGION202.ENABLE

Controls if region 202 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION202.EXEMPT

Mark IDAU region202 as exempt

Type

bool

Default value

0x0

IDAU_REGION202.LADDR

Limit address of IDAU region202

Type

int

Default value

0x0

IDAU_REGION202.NSC

Controls if region 202 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION203.BADDR

Base address of IDAU region203

Type

int

Default value

0x0

IDAU_REGION203.ENABLE

Controls if region 203 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION203.EXEMPT

Mark IDAU region203 as exempt

Type

bool

Default value

0x0

IDAU_REGION203.LADDR

Limit address of IDAU region203

Type

int

Default value

0x0

IDAU_REGION203.NSC

Controls if region 203 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION204.BADDR

Base address of IDAU region204

Type

int

Default value

0x0

IDAU_REGION204.ENABLE

Controls if region 204 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION204.EXEMPT

Mark IDAU region204 as exempt

Type

bool

Default value

0x0

IDAU_REGION204.LADDR

Limit address of IDAU region204

Type

int

Default value

0x0

IDAU_REGION204.NSC

Controls if region 204 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION205.BADDR

Base address of IDAU region205

Type

int

Default value

0x0

IDAU_REGION205.ENABLE

Controls if region 205 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION205.EXEMPT

Mark IDAU region205 as exempt

Type

bool

Default value

0x0

IDAU_REGION205.LADDR

Limit address of IDAU region205

Type

int

Default value

0x0

IDAU_REGION205.NSC

Controls if region 205 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION206.BADDR

Base address of IDAU region206

Type

int

Default value

0x0

IDAU_REGION206.ENABLE

Controls if region 206 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION206.EXEMPT

Mark IDAU region206 as exempt

Type

bool

Default value

0x0

IDAU_REGION206.LADDR

Limit address of IDAU region206

Type

int

Default value

0x0

IDAU_REGION206.NSC

Controls if region 206 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION207.BADDR

Base address of IDAU region207

Type

int

Default value

0x0

IDAU_REGION207.ENABLE

Controls if region 207 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION207.EXEMPT

Mark IDAU region207 as exempt

Type

bool

Default value

0x0

IDAU_REGION207.LADDR

Limit address of IDAU region207

Type

int

Default value

0x0

IDAU_REGION207.NSC

Controls if region 207 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION208.BADDR

Base address of IDAU region208

Type

int

Default value

0x0

IDAU_REGION208.ENABLE

Controls if region 208 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION208.EXEMPT

Mark IDAU region208 as exempt

Type

bool

Default value

0x0

IDAU_REGION208.LADDR

Limit address of IDAU region208

Type

int

Default value

0x0

IDAU_REGION208.NSC

Controls if region 208 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION209.BADDR

Base address of IDAU region209

Type

int

Default value

0x0

IDAU_REGION209.ENABLE

Controls if region 209 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION209.EXEMPT

Mark IDAU region209 as exempt

Type

bool

Default value

0x0

IDAU_REGION209.LADDR

Limit address of IDAU region209

Type

int

Default value

0x0

IDAU_REGION209.NSC

Controls if region 209 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION21.BADDR

Base address of IDAU region21

Type

int

Default value

0x0

IDAU_REGION21.ENABLE

Controls if region 21 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION21.EXEMPT

Mark IDAU region21 as exempt

Type

bool

Default value

0x0

IDAU_REGION21.LADDR

Limit address of IDAU region21

Type

int

Default value

0x0

IDAU_REGION21.NSC

Controls if region 21 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION210.BADDR

Base address of IDAU region210

Type

int

Default value

0x0

IDAU_REGION210.ENABLE

Controls if region 210 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION210.EXEMPT

Mark IDAU region210 as exempt

Type

bool

Default value

0x0

IDAU_REGION210.LADDR

Limit address of IDAU region210

Type

int

Default value

0x0

IDAU_REGION210.NSC

Controls if region 210 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION211.BADDR

Base address of IDAU region211

Type

int

Default value

0x0

IDAU_REGION211.ENABLE

Controls if region 211 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION211.EXEMPT

Mark IDAU region211 as exempt

Type

bool

Default value

0x0

IDAU_REGION211.LADDR

Limit address of IDAU region211

Type

int

Default value

0x0

IDAU_REGION211.NSC

Controls if region 211 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION212.BADDR

Base address of IDAU region212

Type

int

Default value

0x0

IDAU_REGION212.ENABLE

Controls if region 212 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION212.EXEMPT

Mark IDAU region212 as exempt

Type

bool

Default value

0x0

IDAU_REGION212.LADDR

Limit address of IDAU region212

Type

int

Default value

0x0

IDAU_REGION212.NSC

Controls if region 212 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION213.BADDR

Base address of IDAU region213

Type

int

Default value

0x0

IDAU_REGION213.ENABLE

Controls if region 213 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION213.EXEMPT

Mark IDAU region213 as exempt

Type

bool

Default value

0x0

IDAU_REGION213.LADDR

Limit address of IDAU region213

Type

int

Default value

0x0

IDAU_REGION213.NSC

Controls if region 213 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION214.BADDR

Base address of IDAU region214

Type

int

Default value

0x0

IDAU_REGION214.ENABLE

Controls if region 214 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION214.EXEMPT

Mark IDAU region214 as exempt

Type

bool

Default value

0x0

IDAU_REGION214.LADDR

Limit address of IDAU region214

Type

int

Default value

0x0

IDAU_REGION214.NSC

Controls if region 214 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION215.BADDR

Base address of IDAU region215

Type

int

Default value

0x0

IDAU_REGION215.ENABLE

Controls if region 215 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION215.EXEMPT

Mark IDAU region215 as exempt

Type

bool

Default value

0x0

IDAU_REGION215.LADDR

Limit address of IDAU region215

Type

int

Default value

0x0

IDAU_REGION215.NSC

Controls if region 215 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION216.BADDR

Base address of IDAU region216

Type

int

Default value

0x0

IDAU_REGION216.ENABLE

Controls if region 216 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION216.EXEMPT

Mark IDAU region216 as exempt

Type

bool

Default value

0x0

IDAU_REGION216.LADDR

Limit address of IDAU region216

Type

int

Default value

0x0

IDAU_REGION216.NSC

Controls if region 216 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION217.BADDR

Base address of IDAU region217

Type

int

Default value

0x0

IDAU_REGION217.ENABLE

Controls if region 217 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION217.EXEMPT

Mark IDAU region217 as exempt

Type

bool

Default value

0x0

IDAU_REGION217.LADDR

Limit address of IDAU region217

Type

int

Default value

0x0

IDAU_REGION217.NSC

Controls if region 217 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION218.BADDR

Base address of IDAU region218

Type

int

Default value

0x0

IDAU_REGION218.ENABLE

Controls if region 218 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION218.EXEMPT

Mark IDAU region218 as exempt

Type

bool

Default value

0x0

IDAU_REGION218.LADDR

Limit address of IDAU region218

Type

int

Default value

0x0

IDAU_REGION218.NSC

Controls if region 218 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION219.BADDR

Base address of IDAU region219

Type

int

Default value

0x0

IDAU_REGION219.ENABLE

Controls if region 219 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION219.EXEMPT

Mark IDAU region219 as exempt

Type

bool

Default value

0x0

IDAU_REGION219.LADDR

Limit address of IDAU region219

Type

int

Default value

0x0

IDAU_REGION219.NSC

Controls if region 219 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION22.BADDR

Base address of IDAU region22

Type

int

Default value

0x0

IDAU_REGION22.ENABLE

Controls if region 22 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION22.EXEMPT

Mark IDAU region22 as exempt

Type

bool

Default value

0x0

IDAU_REGION22.LADDR

Limit address of IDAU region22

Type

int

Default value

0x0

IDAU_REGION22.NSC

Controls if region 22 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION220.BADDR

Base address of IDAU region220

Type

int

Default value

0x0

IDAU_REGION220.ENABLE

Controls if region 220 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION220.EXEMPT

Mark IDAU region220 as exempt

Type

bool

Default value

0x0

IDAU_REGION220.LADDR

Limit address of IDAU region220

Type

int

Default value

0x0

IDAU_REGION220.NSC

Controls if region 220 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION221.BADDR

Base address of IDAU region221

Type

int

Default value

0x0

IDAU_REGION221.ENABLE

Controls if region 221 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION221.EXEMPT

Mark IDAU region221 as exempt

Type

bool

Default value

0x0

IDAU_REGION221.LADDR

Limit address of IDAU region221

Type

int

Default value

0x0

IDAU_REGION221.NSC

Controls if region 221 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION222.BADDR

Base address of IDAU region222

Type

int

Default value

0x0

IDAU_REGION222.ENABLE

Controls if region 222 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION222.EXEMPT

Mark IDAU region222 as exempt

Type

bool

Default value

0x0

IDAU_REGION222.LADDR

Limit address of IDAU region222

Type

int

Default value

0x0

IDAU_REGION222.NSC

Controls if region 222 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION223.BADDR

Base address of IDAU region223

Type

int

Default value

0x0

IDAU_REGION223.ENABLE

Controls if region 223 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION223.EXEMPT

Mark IDAU region223 as exempt

Type

bool

Default value

0x0

IDAU_REGION223.LADDR

Limit address of IDAU region223

Type

int

Default value

0x0

IDAU_REGION223.NSC

Controls if region 223 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION224.BADDR

Base address of IDAU region224

Type

int

Default value

0x0

IDAU_REGION224.ENABLE

Controls if region 224 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION224.EXEMPT

Mark IDAU region224 as exempt

Type

bool

Default value

0x0

IDAU_REGION224.LADDR

Limit address of IDAU region224

Type

int

Default value

0x0

IDAU_REGION224.NSC

Controls if region 224 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION225.BADDR

Base address of IDAU region225

Type

int

Default value

0x0

IDAU_REGION225.ENABLE

Controls if region 225 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION225.EXEMPT

Mark IDAU region225 as exempt

Type

bool

Default value

0x0

IDAU_REGION225.LADDR

Limit address of IDAU region225

Type

int

Default value

0x0

IDAU_REGION225.NSC

Controls if region 225 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION226.BADDR

Base address of IDAU region226

Type

int

Default value

0x0

IDAU_REGION226.ENABLE

Controls if region 226 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION226.EXEMPT

Mark IDAU region226 as exempt

Type

bool

Default value

0x0

IDAU_REGION226.LADDR

Limit address of IDAU region226

Type

int

Default value

0x0

IDAU_REGION226.NSC

Controls if region 226 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION227.BADDR

Base address of IDAU region227

Type

int

Default value

0x0

IDAU_REGION227.ENABLE

Controls if region 227 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION227.EXEMPT

Mark IDAU region227 as exempt

Type

bool

Default value

0x0

IDAU_REGION227.LADDR

Limit address of IDAU region227

Type

int

Default value

0x0

IDAU_REGION227.NSC

Controls if region 227 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION228.BADDR

Base address of IDAU region228

Type

int

Default value

0x0

IDAU_REGION228.ENABLE

Controls if region 228 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION228.EXEMPT

Mark IDAU region228 as exempt

Type

bool

Default value

0x0

IDAU_REGION228.LADDR

Limit address of IDAU region228

Type

int

Default value

0x0

IDAU_REGION228.NSC

Controls if region 228 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION229.BADDR

Base address of IDAU region229

Type

int

Default value

0x0

IDAU_REGION229.ENABLE

Controls if region 229 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION229.EXEMPT

Mark IDAU region229 as exempt

Type

bool

Default value

0x0

IDAU_REGION229.LADDR

Limit address of IDAU region229

Type

int

Default value

0x0

IDAU_REGION229.NSC

Controls if region 229 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION23.BADDR

Base address of IDAU region23

Type

int

Default value

0x0

IDAU_REGION23.ENABLE

Controls if region 23 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION23.EXEMPT

Mark IDAU region23 as exempt

Type

bool

Default value

0x0

IDAU_REGION23.LADDR

Limit address of IDAU region23

Type

int

Default value

0x0

IDAU_REGION23.NSC

Controls if region 23 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION230.BADDR

Base address of IDAU region230

Type

int

Default value

0x0

IDAU_REGION230.ENABLE

Controls if region 230 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION230.EXEMPT

Mark IDAU region230 as exempt

Type

bool

Default value

0x0

IDAU_REGION230.LADDR

Limit address of IDAU region230

Type

int

Default value

0x0

IDAU_REGION230.NSC

Controls if region 230 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION231.BADDR

Base address of IDAU region231

Type

int

Default value

0x0

IDAU_REGION231.ENABLE

Controls if region 231 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION231.EXEMPT

Mark IDAU region231 as exempt

Type

bool

Default value

0x0

IDAU_REGION231.LADDR

Limit address of IDAU region231

Type

int

Default value

0x0

IDAU_REGION231.NSC

Controls if region 231 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION232.BADDR

Base address of IDAU region232

Type

int

Default value

0x0

IDAU_REGION232.ENABLE

Controls if region 232 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION232.EXEMPT

Mark IDAU region232 as exempt

Type

bool

Default value

0x0

IDAU_REGION232.LADDR

Limit address of IDAU region232

Type

int

Default value

0x0

IDAU_REGION232.NSC

Controls if region 232 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION233.BADDR

Base address of IDAU region233

Type

int

Default value

0x0

IDAU_REGION233.ENABLE

Controls if region 233 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION233.EXEMPT

Mark IDAU region233 as exempt

Type

bool

Default value

0x0

IDAU_REGION233.LADDR

Limit address of IDAU region233

Type

int

Default value

0x0

IDAU_REGION233.NSC

Controls if region 233 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION234.BADDR

Base address of IDAU region234

Type

int

Default value

0x0

IDAU_REGION234.ENABLE

Controls if region 234 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION234.EXEMPT

Mark IDAU region234 as exempt

Type

bool

Default value

0x0

IDAU_REGION234.LADDR

Limit address of IDAU region234

Type

int

Default value

0x0

IDAU_REGION234.NSC

Controls if region 234 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION235.BADDR

Base address of IDAU region235

Type

int

Default value

0x0

IDAU_REGION235.ENABLE

Controls if region 235 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION235.EXEMPT

Mark IDAU region235 as exempt

Type

bool

Default value

0x0

IDAU_REGION235.LADDR

Limit address of IDAU region235

Type

int

Default value

0x0

IDAU_REGION235.NSC

Controls if region 235 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION236.BADDR

Base address of IDAU region236

Type

int

Default value

0x0

IDAU_REGION236.ENABLE

Controls if region 236 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION236.EXEMPT

Mark IDAU region236 as exempt

Type

bool

Default value

0x0

IDAU_REGION236.LADDR

Limit address of IDAU region236

Type

int

Default value

0x0

IDAU_REGION236.NSC

Controls if region 236 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION237.BADDR

Base address of IDAU region237

Type

int

Default value

0x0

IDAU_REGION237.ENABLE

Controls if region 237 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION237.EXEMPT

Mark IDAU region237 as exempt

Type

bool

Default value

0x0

IDAU_REGION237.LADDR

Limit address of IDAU region237

Type

int

Default value

0x0

IDAU_REGION237.NSC

Controls if region 237 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION238.BADDR

Base address of IDAU region238

Type

int

Default value

0x0

IDAU_REGION238.ENABLE

Controls if region 238 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION238.EXEMPT

Mark IDAU region238 as exempt

Type

bool

Default value

0x0

IDAU_REGION238.LADDR

Limit address of IDAU region238

Type

int

Default value

0x0

IDAU_REGION238.NSC

Controls if region 238 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION239.BADDR

Base address of IDAU region239

Type

int

Default value

0x0

IDAU_REGION239.ENABLE

Controls if region 239 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION239.EXEMPT

Mark IDAU region239 as exempt

Type

bool

Default value

0x0

IDAU_REGION239.LADDR

Limit address of IDAU region239

Type

int

Default value

0x0

IDAU_REGION239.NSC

Controls if region 239 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION24.BADDR

Base address of IDAU region24

Type

int

Default value

0x0

IDAU_REGION24.ENABLE

Controls if region 24 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION24.EXEMPT

Mark IDAU region24 as exempt

Type

bool

Default value

0x0

IDAU_REGION24.LADDR

Limit address of IDAU region24

Type

int

Default value

0x0

IDAU_REGION24.NSC

Controls if region 24 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION240.BADDR

Base address of IDAU region240

Type

int

Default value

0x0

IDAU_REGION240.ENABLE

Controls if region 240 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION240.EXEMPT

Mark IDAU region240 as exempt

Type

bool

Default value

0x0

IDAU_REGION240.LADDR

Limit address of IDAU region240

Type

int

Default value

0x0

IDAU_REGION240.NSC

Controls if region 240 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION241.BADDR

Base address of IDAU region241

Type

int

Default value

0x0

IDAU_REGION241.ENABLE

Controls if region 241 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION241.EXEMPT

Mark IDAU region241 as exempt

Type

bool

Default value

0x0

IDAU_REGION241.LADDR

Limit address of IDAU region241

Type

int

Default value

0x0

IDAU_REGION241.NSC

Controls if region 241 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION242.BADDR

Base address of IDAU region242

Type

int

Default value

0x0

IDAU_REGION242.ENABLE

Controls if region 242 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION242.EXEMPT

Mark IDAU region242 as exempt

Type

bool

Default value

0x0

IDAU_REGION242.LADDR

Limit address of IDAU region242

Type

int

Default value

0x0

IDAU_REGION242.NSC

Controls if region 242 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION243.BADDR

Base address of IDAU region243

Type

int

Default value

0x0

IDAU_REGION243.ENABLE

Controls if region 243 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION243.EXEMPT

Mark IDAU region243 as exempt

Type

bool

Default value

0x0

IDAU_REGION243.LADDR

Limit address of IDAU region243

Type

int

Default value

0x0

IDAU_REGION243.NSC

Controls if region 243 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION244.BADDR

Base address of IDAU region244

Type

int

Default value

0x0

IDAU_REGION244.ENABLE

Controls if region 244 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION244.EXEMPT

Mark IDAU region244 as exempt

Type

bool

Default value

0x0

IDAU_REGION244.LADDR

Limit address of IDAU region244

Type

int

Default value

0x0

IDAU_REGION244.NSC

Controls if region 244 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION245.BADDR

Base address of IDAU region245

Type

int

Default value

0x0

IDAU_REGION245.ENABLE

Controls if region 245 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION245.EXEMPT

Mark IDAU region245 as exempt

Type

bool

Default value

0x0

IDAU_REGION245.LADDR

Limit address of IDAU region245

Type

int

Default value

0x0

IDAU_REGION245.NSC

Controls if region 245 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION246.BADDR

Base address of IDAU region246

Type

int

Default value

0x0

IDAU_REGION246.ENABLE

Controls if region 246 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION246.EXEMPT

Mark IDAU region246 as exempt

Type

bool

Default value

0x0

IDAU_REGION246.LADDR

Limit address of IDAU region246

Type

int

Default value

0x0

IDAU_REGION246.NSC

Controls if region 246 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION247.BADDR

Base address of IDAU region247

Type

int

Default value

0x0

IDAU_REGION247.ENABLE

Controls if region 247 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION247.EXEMPT

Mark IDAU region247 as exempt

Type

bool

Default value

0x0

IDAU_REGION247.LADDR

Limit address of IDAU region247

Type

int

Default value

0x0

IDAU_REGION247.NSC

Controls if region 247 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION248.BADDR

Base address of IDAU region248

Type

int

Default value

0x0

IDAU_REGION248.ENABLE

Controls if region 248 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION248.EXEMPT

Mark IDAU region248 as exempt

Type

bool

Default value

0x0

IDAU_REGION248.LADDR

Limit address of IDAU region248

Type

int

Default value

0x0

IDAU_REGION248.NSC

Controls if region 248 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION249.BADDR

Base address of IDAU region249

Type

int

Default value

0x0

IDAU_REGION249.ENABLE

Controls if region 249 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION249.EXEMPT

Mark IDAU region249 as exempt

Type

bool

Default value

0x0

IDAU_REGION249.LADDR

Limit address of IDAU region249

Type

int

Default value

0x0

IDAU_REGION249.NSC

Controls if region 249 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION25.BADDR

Base address of IDAU region25

Type

int

Default value

0x0

IDAU_REGION25.ENABLE

Controls if region 25 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION25.EXEMPT

Mark IDAU region25 as exempt

Type

bool

Default value

0x0

IDAU_REGION25.LADDR

Limit address of IDAU region25

Type

int

Default value

0x0

IDAU_REGION25.NSC

Controls if region 25 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION250.BADDR

Base address of IDAU region250

Type

int

Default value

0x0

IDAU_REGION250.ENABLE

Controls if region 250 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION250.EXEMPT

Mark IDAU region250 as exempt

Type

bool

Default value

0x0

IDAU_REGION250.LADDR

Limit address of IDAU region250

Type

int

Default value

0x0

IDAU_REGION250.NSC

Controls if region 250 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION251.BADDR

Base address of IDAU region251

Type

int

Default value

0x0

IDAU_REGION251.ENABLE

Controls if region 251 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION251.EXEMPT

Mark IDAU region251 as exempt

Type

bool

Default value

0x0

IDAU_REGION251.LADDR

Limit address of IDAU region251

Type

int

Default value

0x0

IDAU_REGION251.NSC

Controls if region 251 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION252.BADDR

Base address of IDAU region252

Type

int

Default value

0x0

IDAU_REGION252.ENABLE

Controls if region 252 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION252.EXEMPT

Mark IDAU region252 as exempt

Type

bool

Default value

0x0

IDAU_REGION252.LADDR

Limit address of IDAU region252

Type

int

Default value

0x0

IDAU_REGION252.NSC

Controls if region 252 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION253.BADDR

Base address of IDAU region253

Type

int

Default value

0x0

IDAU_REGION253.ENABLE

Controls if region 253 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION253.EXEMPT

Mark IDAU region253 as exempt

Type

bool

Default value

0x0

IDAU_REGION253.LADDR

Limit address of IDAU region253

Type

int

Default value

0x0

IDAU_REGION253.NSC

Controls if region 253 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION254.BADDR

Base address of IDAU region254

Type

int

Default value

0x0

IDAU_REGION254.ENABLE

Controls if region 254 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION254.EXEMPT

Mark IDAU region254 as exempt

Type

bool

Default value

0x0

IDAU_REGION254.LADDR

Limit address of IDAU region254

Type

int

Default value

0x0

IDAU_REGION254.NSC

Controls if region 254 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION255.BADDR

Base address of IDAU region255

Type

int

Default value

0x0

IDAU_REGION255.ENABLE

Controls if region 255 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION255.EXEMPT

Mark IDAU region255 as exempt

Type

bool

Default value

0x0

IDAU_REGION255.LADDR

Limit address of IDAU region255

Type

int

Default value

0x0

IDAU_REGION255.NSC

Controls if region 255 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION26.BADDR

Base address of IDAU region26

Type

int

Default value

0x0

IDAU_REGION26.ENABLE

Controls if region 26 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION26.EXEMPT

Mark IDAU region26 as exempt

Type

bool

Default value

0x0

IDAU_REGION26.LADDR

Limit address of IDAU region26

Type

int

Default value

0x0

IDAU_REGION26.NSC

Controls if region 26 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION27.BADDR

Base address of IDAU region27

Type

int

Default value

0x0

IDAU_REGION27.ENABLE

Controls if region 27 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION27.EXEMPT

Mark IDAU region27 as exempt

Type

bool

Default value

0x0

IDAU_REGION27.LADDR

Limit address of IDAU region27

Type

int

Default value

0x0

IDAU_REGION27.NSC

Controls if region 27 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION28.BADDR

Base address of IDAU region28

Type

int

Default value

0x0

IDAU_REGION28.ENABLE

Controls if region 28 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION28.EXEMPT

Mark IDAU region28 as exempt

Type

bool

Default value

0x0

IDAU_REGION28.LADDR

Limit address of IDAU region28

Type

int

Default value

0x0

IDAU_REGION28.NSC

Controls if region 28 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION29.BADDR

Base address of IDAU region29

Type

int

Default value

0x0

IDAU_REGION29.ENABLE

Controls if region 29 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION29.EXEMPT

Mark IDAU region29 as exempt

Type

bool

Default value

0x0

IDAU_REGION29.LADDR

Limit address of IDAU region29

Type

int

Default value

0x0

IDAU_REGION29.NSC

Controls if region 29 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION3.BADDR

Base address of IDAU region3

Type

int

Default value

0x0

IDAU_REGION3.ENABLE

Controls if region 3 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION3.EXEMPT

Mark IDAU region3 as exempt

Type

bool

Default value

0x0

IDAU_REGION3.LADDR

Limit address of IDAU region3

Type

int

Default value

0x0

IDAU_REGION3.NSC

Controls if region 3 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION30.BADDR

Base address of IDAU region30

Type

int

Default value

0x0

IDAU_REGION30.ENABLE

Controls if region 30 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION30.EXEMPT

Mark IDAU region30 as exempt

Type

bool

Default value

0x0

IDAU_REGION30.LADDR

Limit address of IDAU region30

Type

int

Default value

0x0

IDAU_REGION30.NSC

Controls if region 30 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION31.BADDR

Base address of IDAU region31

Type

int

Default value

0x0

IDAU_REGION31.ENABLE

Controls if region 31 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION31.EXEMPT

Mark IDAU region31 as exempt

Type

bool

Default value

0x0

IDAU_REGION31.LADDR

Limit address of IDAU region31

Type

int

Default value

0x0

IDAU_REGION31.NSC

Controls if region 31 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION32.BADDR

Base address of IDAU region32

Type

int

Default value

0x0

IDAU_REGION32.ENABLE

Controls if region 32 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION32.EXEMPT

Mark IDAU region32 as exempt

Type

bool

Default value

0x0

IDAU_REGION32.LADDR

Limit address of IDAU region32

Type

int

Default value

0x0

IDAU_REGION32.NSC

Controls if region 32 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION33.BADDR

Base address of IDAU region33

Type

int

Default value

0x0

IDAU_REGION33.ENABLE

Controls if region 33 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION33.EXEMPT

Mark IDAU region33 as exempt

Type

bool

Default value

0x0

IDAU_REGION33.LADDR

Limit address of IDAU region33

Type

int

Default value

0x0

IDAU_REGION33.NSC

Controls if region 33 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION34.BADDR

Base address of IDAU region34

Type

int

Default value

0x0

IDAU_REGION34.ENABLE

Controls if region 34 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION34.EXEMPT

Mark IDAU region34 as exempt

Type

bool

Default value

0x0

IDAU_REGION34.LADDR

Limit address of IDAU region34

Type

int

Default value

0x0

IDAU_REGION34.NSC

Controls if region 34 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION35.BADDR

Base address of IDAU region35

Type

int

Default value

0x0

IDAU_REGION35.ENABLE

Controls if region 35 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION35.EXEMPT

Mark IDAU region35 as exempt

Type

bool

Default value

0x0

IDAU_REGION35.LADDR

Limit address of IDAU region35

Type

int

Default value

0x0

IDAU_REGION35.NSC

Controls if region 35 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION36.BADDR

Base address of IDAU region36

Type

int

Default value

0x0

IDAU_REGION36.ENABLE

Controls if region 36 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION36.EXEMPT

Mark IDAU region36 as exempt

Type

bool

Default value

0x0

IDAU_REGION36.LADDR

Limit address of IDAU region36

Type

int

Default value

0x0

IDAU_REGION36.NSC

Controls if region 36 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION37.BADDR

Base address of IDAU region37

Type

int

Default value

0x0

IDAU_REGION37.ENABLE

Controls if region 37 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION37.EXEMPT

Mark IDAU region37 as exempt

Type

bool

Default value

0x0

IDAU_REGION37.LADDR

Limit address of IDAU region37

Type

int

Default value

0x0

IDAU_REGION37.NSC

Controls if region 37 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION38.BADDR

Base address of IDAU region38

Type

int

Default value

0x0

IDAU_REGION38.ENABLE

Controls if region 38 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION38.EXEMPT

Mark IDAU region38 as exempt

Type

bool

Default value

0x0

IDAU_REGION38.LADDR

Limit address of IDAU region38

Type

int

Default value

0x0

IDAU_REGION38.NSC

Controls if region 38 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION39.BADDR

Base address of IDAU region39

Type

int

Default value

0x0

IDAU_REGION39.ENABLE

Controls if region 39 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION39.EXEMPT

Mark IDAU region39 as exempt

Type

bool

Default value

0x0

IDAU_REGION39.LADDR

Limit address of IDAU region39

Type

int

Default value

0x0

IDAU_REGION39.NSC

Controls if region 39 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION4.BADDR

Base address of IDAU region4

Type

int

Default value

0x0

IDAU_REGION4.ENABLE

Controls if region 4 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION4.EXEMPT

Mark IDAU region4 as exempt

Type

bool

Default value

0x0

IDAU_REGION4.LADDR

Limit address of IDAU region4

Type

int

Default value

0x0

IDAU_REGION4.NSC

Controls if region 4 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION40.BADDR

Base address of IDAU region40

Type

int

Default value

0x0

IDAU_REGION40.ENABLE

Controls if region 40 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION40.EXEMPT

Mark IDAU region40 as exempt

Type

bool

Default value

0x0

IDAU_REGION40.LADDR

Limit address of IDAU region40

Type

int

Default value

0x0

IDAU_REGION40.NSC

Controls if region 40 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION41.BADDR

Base address of IDAU region41

Type

int

Default value

0x0

IDAU_REGION41.ENABLE

Controls if region 41 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION41.EXEMPT

Mark IDAU region41 as exempt

Type

bool

Default value

0x0

IDAU_REGION41.LADDR

Limit address of IDAU region41

Type

int

Default value

0x0

IDAU_REGION41.NSC

Controls if region 41 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION42.BADDR

Base address of IDAU region42

Type

int

Default value

0x0

IDAU_REGION42.ENABLE

Controls if region 42 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION42.EXEMPT

Mark IDAU region42 as exempt

Type

bool

Default value

0x0

IDAU_REGION42.LADDR

Limit address of IDAU region42

Type

int

Default value

0x0

IDAU_REGION42.NSC

Controls if region 42 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION43.BADDR

Base address of IDAU region43

Type

int

Default value

0x0

IDAU_REGION43.ENABLE

Controls if region 43 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION43.EXEMPT

Mark IDAU region43 as exempt

Type

bool

Default value

0x0

IDAU_REGION43.LADDR

Limit address of IDAU region43

Type

int

Default value

0x0

IDAU_REGION43.NSC

Controls if region 43 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION44.BADDR

Base address of IDAU region44

Type

int

Default value

0x0

IDAU_REGION44.ENABLE

Controls if region 44 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION44.EXEMPT

Mark IDAU region44 as exempt

Type

bool

Default value

0x0

IDAU_REGION44.LADDR

Limit address of IDAU region44

Type

int

Default value

0x0

IDAU_REGION44.NSC

Controls if region 44 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION45.BADDR

Base address of IDAU region45

Type

int

Default value

0x0

IDAU_REGION45.ENABLE

Controls if region 45 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION45.EXEMPT

Mark IDAU region45 as exempt

Type

bool

Default value

0x0

IDAU_REGION45.LADDR

Limit address of IDAU region45

Type

int

Default value

0x0

IDAU_REGION45.NSC

Controls if region 45 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION46.BADDR

Base address of IDAU region46

Type

int

Default value

0x0

IDAU_REGION46.ENABLE

Controls if region 46 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION46.EXEMPT

Mark IDAU region46 as exempt

Type

bool

Default value

0x0

IDAU_REGION46.LADDR

Limit address of IDAU region46

Type

int

Default value

0x0

IDAU_REGION46.NSC

Controls if region 46 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION47.BADDR

Base address of IDAU region47

Type

int

Default value

0x0

IDAU_REGION47.ENABLE

Controls if region 47 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION47.EXEMPT

Mark IDAU region47 as exempt

Type

bool

Default value

0x0

IDAU_REGION47.LADDR

Limit address of IDAU region47

Type

int

Default value

0x0

IDAU_REGION47.NSC

Controls if region 47 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION48.BADDR

Base address of IDAU region48

Type

int

Default value

0x0

IDAU_REGION48.ENABLE

Controls if region 48 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION48.EXEMPT

Mark IDAU region48 as exempt

Type

bool

Default value

0x0

IDAU_REGION48.LADDR

Limit address of IDAU region48

Type

int

Default value

0x0

IDAU_REGION48.NSC

Controls if region 48 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION49.BADDR

Base address of IDAU region49

Type

int

Default value

0x0

IDAU_REGION49.ENABLE

Controls if region 49 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION49.EXEMPT

Mark IDAU region49 as exempt

Type

bool

Default value

0x0

IDAU_REGION49.LADDR

Limit address of IDAU region49

Type

int

Default value

0x0

IDAU_REGION49.NSC

Controls if region 49 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION5.BADDR

Base address of IDAU region5

Type

int

Default value

0x0

IDAU_REGION5.ENABLE

Controls if region 5 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION5.EXEMPT

Mark IDAU region5 as exempt

Type

bool

Default value

0x0

IDAU_REGION5.LADDR

Limit address of IDAU region5

Type

int

Default value

0x0

IDAU_REGION5.NSC

Controls if region 5 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION50.BADDR

Base address of IDAU region50

Type

int

Default value

0x0

IDAU_REGION50.ENABLE

Controls if region 50 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION50.EXEMPT

Mark IDAU region50 as exempt

Type

bool

Default value

0x0

IDAU_REGION50.LADDR

Limit address of IDAU region50

Type

int

Default value

0x0

IDAU_REGION50.NSC

Controls if region 50 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION51.BADDR

Base address of IDAU region51

Type

int

Default value

0x0

IDAU_REGION51.ENABLE

Controls if region 51 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION51.EXEMPT

Mark IDAU region51 as exempt

Type

bool

Default value

0x0

IDAU_REGION51.LADDR

Limit address of IDAU region51

Type

int

Default value

0x0

IDAU_REGION51.NSC

Controls if region 51 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION52.BADDR

Base address of IDAU region52

Type

int

Default value

0x0

IDAU_REGION52.ENABLE

Controls if region 52 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION52.EXEMPT

Mark IDAU region52 as exempt

Type

bool

Default value

0x0

IDAU_REGION52.LADDR

Limit address of IDAU region52

Type

int

Default value

0x0

IDAU_REGION52.NSC

Controls if region 52 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION53.BADDR

Base address of IDAU region53

Type

int

Default value

0x0

IDAU_REGION53.ENABLE

Controls if region 53 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION53.EXEMPT

Mark IDAU region53 as exempt

Type

bool

Default value

0x0

IDAU_REGION53.LADDR

Limit address of IDAU region53

Type

int

Default value

0x0

IDAU_REGION53.NSC

Controls if region 53 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION54.BADDR

Base address of IDAU region54

Type

int

Default value

0x0

IDAU_REGION54.ENABLE

Controls if region 54 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION54.EXEMPT

Mark IDAU region54 as exempt

Type

bool

Default value

0x0

IDAU_REGION54.LADDR

Limit address of IDAU region54

Type

int

Default value

0x0

IDAU_REGION54.NSC

Controls if region 54 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION55.BADDR

Base address of IDAU region55

Type

int

Default value

0x0

IDAU_REGION55.ENABLE

Controls if region 55 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION55.EXEMPT

Mark IDAU region55 as exempt

Type

bool

Default value

0x0

IDAU_REGION55.LADDR

Limit address of IDAU region55

Type

int

Default value

0x0

IDAU_REGION55.NSC

Controls if region 55 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION56.BADDR

Base address of IDAU region56

Type

int

Default value

0x0

IDAU_REGION56.ENABLE

Controls if region 56 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION56.EXEMPT

Mark IDAU region56 as exempt

Type

bool

Default value

0x0

IDAU_REGION56.LADDR

Limit address of IDAU region56

Type

int

Default value

0x0

IDAU_REGION56.NSC

Controls if region 56 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION57.BADDR

Base address of IDAU region57

Type

int

Default value

0x0

IDAU_REGION57.ENABLE

Controls if region 57 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION57.EXEMPT

Mark IDAU region57 as exempt

Type

bool

Default value

0x0

IDAU_REGION57.LADDR

Limit address of IDAU region57

Type

int

Default value

0x0

IDAU_REGION57.NSC

Controls if region 57 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION58.BADDR

Base address of IDAU region58

Type

int

Default value

0x0

IDAU_REGION58.ENABLE

Controls if region 58 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION58.EXEMPT

Mark IDAU region58 as exempt

Type

bool

Default value

0x0

IDAU_REGION58.LADDR

Limit address of IDAU region58

Type

int

Default value

0x0

IDAU_REGION58.NSC

Controls if region 58 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION59.BADDR

Base address of IDAU region59

Type

int

Default value

0x0

IDAU_REGION59.ENABLE

Controls if region 59 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION59.EXEMPT

Mark IDAU region59 as exempt

Type

bool

Default value

0x0

IDAU_REGION59.LADDR

Limit address of IDAU region59

Type

int

Default value

0x0

IDAU_REGION59.NSC

Controls if region 59 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION6.BADDR

Base address of IDAU region6

Type

int

Default value

0x0

IDAU_REGION6.ENABLE

Controls if region 6 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION6.EXEMPT

Mark IDAU region6 as exempt

Type

bool

Default value

0x0

IDAU_REGION6.LADDR

Limit address of IDAU region6

Type

int

Default value

0x0

IDAU_REGION6.NSC

Controls if region 6 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION60.BADDR

Base address of IDAU region60

Type

int

Default value

0x0

IDAU_REGION60.ENABLE

Controls if region 60 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION60.EXEMPT

Mark IDAU region60 as exempt

Type

bool

Default value

0x0

IDAU_REGION60.LADDR

Limit address of IDAU region60

Type

int

Default value

0x0

IDAU_REGION60.NSC

Controls if region 60 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION61.BADDR

Base address of IDAU region61

Type

int

Default value

0x0

IDAU_REGION61.ENABLE

Controls if region 61 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION61.EXEMPT

Mark IDAU region61 as exempt

Type

bool

Default value

0x0

IDAU_REGION61.LADDR

Limit address of IDAU region61

Type

int

Default value

0x0

IDAU_REGION61.NSC

Controls if region 61 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION62.BADDR

Base address of IDAU region62

Type

int

Default value

0x0

IDAU_REGION62.ENABLE

Controls if region 62 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION62.EXEMPT

Mark IDAU region62 as exempt

Type

bool

Default value

0x0

IDAU_REGION62.LADDR

Limit address of IDAU region62

Type

int

Default value

0x0

IDAU_REGION62.NSC

Controls if region 62 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION63.BADDR

Base address of IDAU region63

Type

int

Default value

0x0

IDAU_REGION63.ENABLE

Controls if region 63 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION63.EXEMPT

Mark IDAU region63 as exempt

Type

bool

Default value

0x0

IDAU_REGION63.LADDR

Limit address of IDAU region63

Type

int

Default value

0x0

IDAU_REGION63.NSC

Controls if region 63 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION64.BADDR

Base address of IDAU region64

Type

int

Default value

0x0

IDAU_REGION64.ENABLE

Controls if region 64 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION64.EXEMPT

Mark IDAU region64 as exempt

Type

bool

Default value

0x0

IDAU_REGION64.LADDR

Limit address of IDAU region64

Type

int

Default value

0x0

IDAU_REGION64.NSC

Controls if region 64 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION65.BADDR

Base address of IDAU region65

Type

int

Default value

0x0

IDAU_REGION65.ENABLE

Controls if region 65 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION65.EXEMPT

Mark IDAU region65 as exempt

Type

bool

Default value

0x0

IDAU_REGION65.LADDR

Limit address of IDAU region65

Type

int

Default value

0x0

IDAU_REGION65.NSC

Controls if region 65 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION66.BADDR

Base address of IDAU region66

Type

int

Default value

0x0

IDAU_REGION66.ENABLE

Controls if region 66 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION66.EXEMPT

Mark IDAU region66 as exempt

Type

bool

Default value

0x0

IDAU_REGION66.LADDR

Limit address of IDAU region66

Type

int

Default value

0x0

IDAU_REGION66.NSC

Controls if region 66 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION67.BADDR

Base address of IDAU region67

Type

int

Default value

0x0

IDAU_REGION67.ENABLE

Controls if region 67 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION67.EXEMPT

Mark IDAU region67 as exempt

Type

bool

Default value

0x0

IDAU_REGION67.LADDR

Limit address of IDAU region67

Type

int

Default value

0x0

IDAU_REGION67.NSC

Controls if region 67 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION68.BADDR

Base address of IDAU region68

Type

int

Default value

0x0

IDAU_REGION68.ENABLE

Controls if region 68 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION68.EXEMPT

Mark IDAU region68 as exempt

Type

bool

Default value

0x0

IDAU_REGION68.LADDR

Limit address of IDAU region68

Type

int

Default value

0x0

IDAU_REGION68.NSC

Controls if region 68 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION69.BADDR

Base address of IDAU region69

Type

int

Default value

0x0

IDAU_REGION69.ENABLE

Controls if region 69 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION69.EXEMPT

Mark IDAU region69 as exempt

Type

bool

Default value

0x0

IDAU_REGION69.LADDR

Limit address of IDAU region69

Type

int

Default value

0x0

IDAU_REGION69.NSC

Controls if region 69 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION7.BADDR

Base address of IDAU region7

Type

int

Default value

0x0

IDAU_REGION7.ENABLE

Controls if region 7 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION7.EXEMPT

Mark IDAU region7 as exempt

Type

bool

Default value

0x0

IDAU_REGION7.LADDR

Limit address of IDAU region7

Type

int

Default value

0x0

IDAU_REGION7.NSC

Controls if region 7 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION70.BADDR

Base address of IDAU region70

Type

int

Default value

0x0

IDAU_REGION70.ENABLE

Controls if region 70 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION70.EXEMPT

Mark IDAU region70 as exempt

Type

bool

Default value

0x0

IDAU_REGION70.LADDR

Limit address of IDAU region70

Type

int

Default value

0x0

IDAU_REGION70.NSC

Controls if region 70 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION71.BADDR

Base address of IDAU region71

Type

int

Default value

0x0

IDAU_REGION71.ENABLE

Controls if region 71 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION71.EXEMPT

Mark IDAU region71 as exempt

Type

bool

Default value

0x0

IDAU_REGION71.LADDR

Limit address of IDAU region71

Type

int

Default value

0x0

IDAU_REGION71.NSC

Controls if region 71 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION72.BADDR

Base address of IDAU region72

Type

int

Default value

0x0

IDAU_REGION72.ENABLE

Controls if region 72 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION72.EXEMPT

Mark IDAU region72 as exempt

Type

bool

Default value

0x0

IDAU_REGION72.LADDR

Limit address of IDAU region72

Type

int

Default value

0x0

IDAU_REGION72.NSC

Controls if region 72 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION73.BADDR

Base address of IDAU region73

Type

int

Default value

0x0

IDAU_REGION73.ENABLE

Controls if region 73 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION73.EXEMPT

Mark IDAU region73 as exempt

Type

bool

Default value

0x0

IDAU_REGION73.LADDR

Limit address of IDAU region73

Type

int

Default value

0x0

IDAU_REGION73.NSC

Controls if region 73 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION74.BADDR

Base address of IDAU region74

Type

int

Default value

0x0

IDAU_REGION74.ENABLE

Controls if region 74 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION74.EXEMPT

Mark IDAU region74 as exempt

Type

bool

Default value

0x0

IDAU_REGION74.LADDR

Limit address of IDAU region74

Type

int

Default value

0x0

IDAU_REGION74.NSC

Controls if region 74 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION75.BADDR

Base address of IDAU region75

Type

int

Default value

0x0

IDAU_REGION75.ENABLE

Controls if region 75 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION75.EXEMPT

Mark IDAU region75 as exempt

Type

bool

Default value

0x0

IDAU_REGION75.LADDR

Limit address of IDAU region75

Type

int

Default value

0x0

IDAU_REGION75.NSC

Controls if region 75 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION76.BADDR

Base address of IDAU region76

Type

int

Default value

0x0

IDAU_REGION76.ENABLE

Controls if region 76 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION76.EXEMPT

Mark IDAU region76 as exempt

Type

bool

Default value

0x0

IDAU_REGION76.LADDR

Limit address of IDAU region76

Type

int

Default value

0x0

IDAU_REGION76.NSC

Controls if region 76 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION77.BADDR

Base address of IDAU region77

Type

int

Default value

0x0

IDAU_REGION77.ENABLE

Controls if region 77 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION77.EXEMPT

Mark IDAU region77 as exempt

Type

bool

Default value

0x0

IDAU_REGION77.LADDR

Limit address of IDAU region77

Type

int

Default value

0x0

IDAU_REGION77.NSC

Controls if region 77 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION78.BADDR

Base address of IDAU region78

Type

int

Default value

0x0

IDAU_REGION78.ENABLE

Controls if region 78 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION78.EXEMPT

Mark IDAU region78 as exempt

Type

bool

Default value

0x0

IDAU_REGION78.LADDR

Limit address of IDAU region78

Type

int

Default value

0x0

IDAU_REGION78.NSC

Controls if region 78 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION79.BADDR

Base address of IDAU region79

Type

int

Default value

0x0

IDAU_REGION79.ENABLE

Controls if region 79 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION79.EXEMPT

Mark IDAU region79 as exempt

Type

bool

Default value

0x0

IDAU_REGION79.LADDR

Limit address of IDAU region79

Type

int

Default value

0x0

IDAU_REGION79.NSC

Controls if region 79 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION8.BADDR

Base address of IDAU region8

Type

int

Default value

0x0

IDAU_REGION8.ENABLE

Controls if region 8 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION8.EXEMPT

Mark IDAU region8 as exempt

Type

bool

Default value

0x0

IDAU_REGION8.LADDR

Limit address of IDAU region8

Type

int

Default value

0x0

IDAU_REGION8.NSC

Controls if region 8 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION80.BADDR

Base address of IDAU region80

Type

int

Default value

0x0

IDAU_REGION80.ENABLE

Controls if region 80 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION80.EXEMPT

Mark IDAU region80 as exempt

Type

bool

Default value

0x0

IDAU_REGION80.LADDR

Limit address of IDAU region80

Type

int

Default value

0x0

IDAU_REGION80.NSC

Controls if region 80 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION81.BADDR

Base address of IDAU region81

Type

int

Default value

0x0

IDAU_REGION81.ENABLE

Controls if region 81 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION81.EXEMPT

Mark IDAU region81 as exempt

Type

bool

Default value

0x0

IDAU_REGION81.LADDR

Limit address of IDAU region81

Type

int

Default value

0x0

IDAU_REGION81.NSC

Controls if region 81 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION82.BADDR

Base address of IDAU region82

Type

int

Default value

0x0

IDAU_REGION82.ENABLE

Controls if region 82 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION82.EXEMPT

Mark IDAU region82 as exempt

Type

bool

Default value

0x0

IDAU_REGION82.LADDR

Limit address of IDAU region82

Type

int

Default value

0x0

IDAU_REGION82.NSC

Controls if region 82 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION83.BADDR

Base address of IDAU region83

Type

int

Default value

0x0

IDAU_REGION83.ENABLE

Controls if region 83 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION83.EXEMPT

Mark IDAU region83 as exempt

Type

bool

Default value

0x0

IDAU_REGION83.LADDR

Limit address of IDAU region83

Type

int

Default value

0x0

IDAU_REGION83.NSC

Controls if region 83 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION84.BADDR

Base address of IDAU region84

Type

int

Default value

0x0

IDAU_REGION84.ENABLE

Controls if region 84 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION84.EXEMPT

Mark IDAU region84 as exempt

Type

bool

Default value

0x0

IDAU_REGION84.LADDR

Limit address of IDAU region84

Type

int

Default value

0x0

IDAU_REGION84.NSC

Controls if region 84 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION85.BADDR

Base address of IDAU region85

Type

int

Default value

0x0

IDAU_REGION85.ENABLE

Controls if region 85 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION85.EXEMPT

Mark IDAU region85 as exempt

Type

bool

Default value

0x0

IDAU_REGION85.LADDR

Limit address of IDAU region85

Type

int

Default value

0x0

IDAU_REGION85.NSC

Controls if region 85 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION86.BADDR

Base address of IDAU region86

Type

int

Default value

0x0

IDAU_REGION86.ENABLE

Controls if region 86 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION86.EXEMPT

Mark IDAU region86 as exempt

Type

bool

Default value

0x0

IDAU_REGION86.LADDR

Limit address of IDAU region86

Type

int

Default value

0x0

IDAU_REGION86.NSC

Controls if region 86 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION87.BADDR

Base address of IDAU region87

Type

int

Default value

0x0

IDAU_REGION87.ENABLE

Controls if region 87 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION87.EXEMPT

Mark IDAU region87 as exempt

Type

bool

Default value

0x0

IDAU_REGION87.LADDR

Limit address of IDAU region87

Type

int

Default value

0x0

IDAU_REGION87.NSC

Controls if region 87 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION88.BADDR

Base address of IDAU region88

Type

int

Default value

0x0

IDAU_REGION88.ENABLE

Controls if region 88 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION88.EXEMPT

Mark IDAU region88 as exempt

Type

bool

Default value

0x0

IDAU_REGION88.LADDR

Limit address of IDAU region88

Type

int

Default value

0x0

IDAU_REGION88.NSC

Controls if region 88 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION89.BADDR

Base address of IDAU region89

Type

int

Default value

0x0

IDAU_REGION89.ENABLE

Controls if region 89 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION89.EXEMPT

Mark IDAU region89 as exempt

Type

bool

Default value

0x0

IDAU_REGION89.LADDR

Limit address of IDAU region89

Type

int

Default value

0x0

IDAU_REGION89.NSC

Controls if region 89 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION9.BADDR

Base address of IDAU region9

Type

int

Default value

0x0

IDAU_REGION9.ENABLE

Controls if region 9 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION9.EXEMPT

Mark IDAU region9 as exempt

Type

bool

Default value

0x0

IDAU_REGION9.LADDR

Limit address of IDAU region9

Type

int

Default value

0x0

IDAU_REGION9.NSC

Controls if region 9 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION90.BADDR

Base address of IDAU region90

Type

int

Default value

0x0

IDAU_REGION90.ENABLE

Controls if region 90 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION90.EXEMPT

Mark IDAU region90 as exempt

Type

bool

Default value

0x0

IDAU_REGION90.LADDR

Limit address of IDAU region90

Type

int

Default value

0x0

IDAU_REGION90.NSC

Controls if region 90 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION91.BADDR

Base address of IDAU region91

Type

int

Default value

0x0

IDAU_REGION91.ENABLE

Controls if region 91 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION91.EXEMPT

Mark IDAU region91 as exempt

Type

bool

Default value

0x0

IDAU_REGION91.LADDR

Limit address of IDAU region91

Type

int

Default value

0x0

IDAU_REGION91.NSC

Controls if region 91 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION92.BADDR

Base address of IDAU region92

Type

int

Default value

0x0

IDAU_REGION92.ENABLE

Controls if region 92 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION92.EXEMPT

Mark IDAU region92 as exempt

Type

bool

Default value

0x0

IDAU_REGION92.LADDR

Limit address of IDAU region92

Type

int

Default value

0x0

IDAU_REGION92.NSC

Controls if region 92 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION93.BADDR

Base address of IDAU region93

Type

int

Default value

0x0

IDAU_REGION93.ENABLE

Controls if region 93 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION93.EXEMPT

Mark IDAU region93 as exempt

Type

bool

Default value

0x0

IDAU_REGION93.LADDR

Limit address of IDAU region93

Type

int

Default value

0x0

IDAU_REGION93.NSC

Controls if region 93 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION94.BADDR

Base address of IDAU region94

Type

int

Default value

0x0

IDAU_REGION94.ENABLE

Controls if region 94 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION94.EXEMPT

Mark IDAU region94 as exempt

Type

bool

Default value

0x0

IDAU_REGION94.LADDR

Limit address of IDAU region94

Type

int

Default value

0x0

IDAU_REGION94.NSC

Controls if region 94 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION95.BADDR

Base address of IDAU region95

Type

int

Default value

0x0

IDAU_REGION95.ENABLE

Controls if region 95 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION95.EXEMPT

Mark IDAU region95 as exempt

Type

bool

Default value

0x0

IDAU_REGION95.LADDR

Limit address of IDAU region95

Type

int

Default value

0x0

IDAU_REGION95.NSC

Controls if region 95 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION96.BADDR

Base address of IDAU region96

Type

int

Default value

0x0

IDAU_REGION96.ENABLE

Controls if region 96 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION96.EXEMPT

Mark IDAU region96 as exempt

Type

bool

Default value

0x0

IDAU_REGION96.LADDR

Limit address of IDAU region96

Type

int

Default value

0x0

IDAU_REGION96.NSC

Controls if region 96 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION97.BADDR

Base address of IDAU region97

Type

int

Default value

0x0

IDAU_REGION97.ENABLE

Controls if region 97 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored. 1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION97.EXEMPT

Mark IDAU region97 as exempt

Type

bool

Default value

0x0

IDAU_REGION97.LADDR

Limit address of IDAU region97

Type

int

Default value

0x0

IDAU_REGION97.NSC

Controls if region 97 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION98.BADDR

Base address of IDAU region98

Type

int

Default value

0x0

IDAU_REGION98.ENABLE

Controls if region 98 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION98.EXEMPT

Mark IDAU region98 as exempt

Type

bool

Default value

0x0

IDAU_REGION98.LADDR

Limit address of IDAU region98

Type

int

Default value

0x0

IDAU_REGION98.NSC

Controls if region 98 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

IDAU_REGION99.BADDR

Base address of IDAU region99

Type

int

Default value

0x0

IDAU_REGION99.ENABLE

Controls if region 99 is S or NS, only valid when NSC=0. If NSC=1 this parameter is ignored.
1 = region is NS, 0 = region is S (absent if LADDR=0)

Type

bool

Default value

0x0

IDAU_REGION99.EXEMPT

Mark IDAU region99 as exempt

Type

bool

Default value

0x0

IDAU_REGION99.LADDR

Limit address of IDAU region99

Type

int

Default value

0x0

IDAU_REGION99.NSC

Controls if region 99 is NSC. 1 = IDAU region is NSC, 0 = ENABLE parameter determines if S or NS.

Type

bool

Default value

0x0

NUM_IDAU_REGION

Number of IDAU regions

Type

int

Default value

0x0

3.10.46 IntegrityChecker

Integrity Checker. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1014: IP revisions support

Revision	Quality level
0.5	Alpha support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- ICDL_CHUNK_SIZE

Iris and MTI instances for IntegrityChecker

This model has the following Iris instances:

Table 3-1015: IntegrityChecker Iris instances

InstanceName	ComponentName
IntegrityChecker	IntegrityChecker
IntegrityChecker.apb	PVBusSlave

This model has the following MTI trace components:

Table 3-1016: IntegrityChecker MTI instances

InstanceName	ComponentName
IntegrityChecker.apb	PVBusSlave

IntegrityChecker contains the following CADI targets:

- IntegrityChecker

Ports for IntegrityChecker

Table 3-1017: Ports

Name	Protocol	Type	Description
apb	PVBus	Slave	APB Subordinate Interface - Access to registers
ic_alarm_out	Signal	Master	Alarm status out signal
ic_interrupt_out	Signal	Master	Interrupt out signal
ic_match_trigger_ack_in	Signal	Slave	IC Match done ack signal

Name	Protocol	Type	Description
ic_match_trigger_req_out	Signal	Master	IC Match done req signal
pvbus_m	PVBus	Master	To read and write to external memory
reset_in	Signal	Slave	Reset in signal
warm_reset_in	Signal	Slave	Warm Reset in signal

Parameters for IntegrityChecker

ICBC_RESET_VALUE

ICBC Registers Reset Value

Type

int

Default value

0x1

ICDL_CHUNK_SIZE

ICDL Chunk Size

Type

int

Default value

0x8

PID0_RESET_VALUE

PID0 Registers Reset Value

Type

int

Default value

0x0

PID1_RESET_VALUE

PID1 Registers Reset Value

Type

int

Default value

0xb0

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type

int

Default value

0x2

3.10.47 KeyManagementUnit

KeyManagementUnit. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1018: IP revisions support

Revision	Quality level
0.5	Alpha support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for KeyManagementUnit

This model has the following Iris instances:

Table 3-1019: KeyManagementUnit Iris instances

InstanceName	ComponentName
KeyManagementUnit	KeyManagementUnit
KeyManagementUnit.apb	PVBusSlave
KeyManagementUnit.keys_in	PVBusSlave

This model has the following MTI trace components:

Table 3-1020: KeyManagementUnit MTI instances

InstanceName	ComponentName
KeyManagementUnit.apb	PVBusSlave
KeyManagementUnit.keys_in	PVBusSlave

KeyManagementUnit contains the following CADI targets:

- KeyManagementUnit

Ports for KeyManagementUnit

Table 3-1021: Ports

Name	Protocol	Type	Description
apb	PVBus	Slave	register access via apb port
hw_keys_in	PVBus	Slave	HW key register access via private apb port - connect to LCM
irq_out	Signal	Master	IRQ signal out
keys_out	PVBus	Master	output keys via port
reset_in	Signal	Slave	Reset signal in

Parameters for KeyManagementUnit

KMUDKPARV0

Reset value of the KMUDKPA<0> register. If no hardware key slot is supported (KMUNHWKSLTS is 0), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUDKPARV1

Reset value of the KMUDKPA<1> register. If no hardware key slot is supported (KMUNHWKSLTS is 1), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUDKPARV2

Reset value of the KMUDKPA<2> register. If no hardware key slot is supported (KMUNHWKSLTS is 2), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUDKPARV3

Reset value of the KMUDKPA<3> register. If no hardware key slot is supported (KMUNHWKSLTS is 3), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUDKPARV4

Reset value of the KMUDKPA<4> register. If no hardware key slot is supported (KMUNHWKSLTS is 4), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUDKPARV5

Reset value of the KMUDKPA<5> register. If no hardware key slot is supported (KMUNHWKSLTS is 5), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUDKPARV6

Reset value of the KMUDKPA<6> register. If no hardware key slot is supported (KMUNHWKSLTS is 6), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUDKPARV7

Reset value of the KMUDKPA<7> register. If no hardware key slot is supported (KMUNHWKSLTS is 7), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUKSCRV0

Reset value of the KMUKSC<0> register. If no hardware key slot is supported (KMUNHWKSLTS is 0), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUKSCRV1

Reset value of the KMUKSC<1> register. If no hardware key slot is supported (KMUNHWKSLTS is 1), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUKSCRV2

Reset value of the KMUKSC<2> register. If no hardware key slot is supported (KMUNHWKSLTS is 2), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUKSCRV3

Reset value of the KMUKSC<3> register. If no hardware key slot is supported (KMUNHWKSLTS is 3), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUKSCRV4

Reset value of the KMUKSC<4> register. If no hardware key slot is supported (KMUNHWKSLTS is 4), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUKSCRV5

Reset value of the KMUKSC<5> register. If no hardware key slot is supported (KMUNHWKSLTS is 5), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUKSCRV6

Reset value of the KMUKSC<6> register. If no hardware key slot is supported (KMUNHWKSLTS is 6), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUKSCRV7

Reset value of the KMUKSC<7> register. If no hardware key slot is supported (KMUNHWKSLTS is 7), this configuration option must be set to 0x0000_0000.

Type

int

Default value

0x0

KMUNHWKSLTS

Number of hardware key slots. Supported values: 0..8 (0 = No hardware key slot). The first KMUNHWKSLTS key slots are filled by hardware (typically by LCM) and the rest are software key slots. (Default=7)

Type

int

Default value

0x7

KMUNKS

Selects the number of key slots that the KMU supports. This value is reflected in KMUBC.NKS. 0x1=2, 0x2=4, 0x3=8, 0x4=16, 0x5=32 key slots (Default=0x5)

Type

int

Default value

0x5

KMUTANG

Specifies four possible values for the Top Address Nibble Groups of the target crypto devices that the KMU allows software to set in the KMUDKPA<n> registers.

Type

int

Default value

0x5

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type

int

Default value

0x2

3.10.48 LifeCycleManager

LifeCycleManager. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1022: IP revisions support

Revision	Quality level
1.02	Alpha support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for LifeCycleManager

This model has the following Iris instances:

Table 3-1023: LifeCycleManager Iris instances

InstanceName	ComponentName
LifeCycleManager	RSS_LifeCycleManager
LifeCycleManager.apb	PVBusSlave
LifeCycleManager.hw_keys_out	PVBusMaster
LifeCycleManager.nvm_external_out	PVBusMaster

This model has the following MTI trace components:

Table 3-1024: LifeCycleManager MTI instances

InstanceName	ComponentName
LifeCycleManager.apb	PVBusSlave
LifeCycleManager.hw_keys_out	PVBusMaster
LifeCycleManager.nvm_external_out	PVBusMaster

LifeCycleManager contains the following CADI targets:

- RSS_LifeCycleManager

Ports for LifeCycleManager

Table 3-1025: Ports

Name	Protocol	Type	Description
apb	PVBus	Slave	APB4 Subordinate Interface - Access to LCM registers and internal OTP/NVM
hw_keys_out	PVBus	Master	APB manager interface for output of the HW Root of Trust keys
lcs	Value	Master	The 3-bit value of the LCS register
nvm_external_out	PVBus	Master	APB manager interface for external NVM (OTP)
reset_in	Signal	Slave	LCM reset in
sp_reset_out	Signal	Master	LCM SP_RST_REQ signal out

Parameters for LifeCycleManager

DCU_SP_DISABLE_MASK_VAL

The Secure Provisioning disable mask of the 128-bit DCU signals. Clearing a bit in the mask will force the relevant DCU signal to zero when LCM is in Secure Provisioning Mode (SP_EN=1).

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

DISABLE_DIRECT_KEY_APB_MASKING

When set to 1, the Direct Key APB masking feature should be disabled.

Type

int

Default value

0x0

KRTL_VAL

The Krtl value. (256-bit hex)

Type

string

Default value

0x7bfee0b730dcc241938458dafc2bc784cc15bdabeb84311f59cca8a9a4e8bc35

OTP_ADDR_WIDTH

The OTP bus width (width in bits = OTP_ADDR_WIDTH + 2)

Type

int

Default value

0xc

OTP_MASK_VAL

This value must be generated using a true random number generator (1536-bit hex)

Type

string

Default value

0x3aaf46ac792c53b0ffd80fe92129f0e8d80a9048799c5498c19f1a1ce7ddf5b20e2cc8f9456cf4a

OTP_SIZE_IN_BYTES

The size of the OTP region accessible from the LCM APB-S interface. The maximum size is 60KB.

Type

int

Default value

0x4000

PCI_DCU_PERMANENT_DISABLE_MASK_VAL_LCS_CM

The permanent disable mask of the 128-bit DCU signals when LCS=CM and TP Mode=PCI. Clearing a bit in the mask will force the relevant DCU signal to zero.

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

PCI_DCU_PERMANENT_DISABLE_MASK_VAL_LCS_DM

The permanent disable mask of the 128-bit DCU signals when LCS=DM and TP Mode=PCI. Clearing a bit in the mask will force the relevant DCU signal to zero.

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

PCI_DCU_PERMANENT_DISABLE_MASK_VAL_LCS_RMA

The permanent disable mask of the 128-bit DCU signals when LCS=RMA and TP Mode=PCI. Clearing a bit in the mask will force the relevant DCU signal to zero.

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

PCI_DCU_PERMANENT_DISABLE_MASK_VAL_LCS_SECURE

The permanent disable mask of the 128-bit DCU signals when LCS=SE and TP Mode=PCI. Clearing a bit in the mask will force the relevant DCU signal to zero.

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

PCI_DEFAULT_DCU_VAL_LCS_CM

The default reset value of the 128-bit DCU signals when LCS=CM and TP Mode=PCI

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

PCI_DEFAULT_DCU_VAL_LCS_DM

The default reset value of the 128-bit DCU signals when LCS=DM and TP Mode=PCI

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

PCI_DEFAULT_DCU_VAL_LCS_RMA

The default reset value of the 128-bit DCU signals when LCS=RMA and TP Mode=PCI

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

PCI_DEFAULT_DCU_VAL_LCS_SECURE

The default reset value of the 128-bit DCU signals when LCS=SE and TP Mode=PCI

Type

string

Default value

0x0

TCI_DCU_PERMANENT_DISABLE_MASK_VAL_LCS_CMThe permanent disable mask of the 128-bit DCU signals when LCS=CM and TP Mode=TCI.
Clearing a bit in the mask will force the relevant DCU signal to zero.**Type**

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

TCI_DCU_PERMANENT_DISABLE_MASK_VAL_LCS_DMThe permanent disable mask of the 128-bit DCU signals when LCS=DM and TP Mode=TCI.
Clearing a bit in the mask will force the relevant DCU signal to zero.**Type**

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

TCI_DCU_PERMANENT_DISABLE_MASK_VAL_LCS_RMA

The permanent disable mask of the 128-bit DCU signals when LCS=RMA and TP Mode=TCI. Clearing a bit in the mask will force the relevant DCU signal to zero.

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

TCI_DCU_PERMANENT_DISABLE_MASK_VAL_LCS_SECURE

The permanent disable mask of the 128-bit DCU signals when LCS=SE and TP Mode=TCI. Clearing a bit in the mask will force the relevant DCU signal to zero.

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

TCI_DEFAULT_DCU_VAL_LCS_CM

The default reset value of the 128-bit DCU signals when LCS=CM and TP Mode=TCI

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

TCI_DEFAULT_DCU_VAL_LCS_DM

The default reset value of the 128-bit DCU signals when LCS=DM and TP Mode=TCI

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

TCI_DEFAULT_DCU_VAL_LCS_RMA

The default reset value of the 128-bit DCU signals when LCS=RMA and TP Mode=TCI

Type

string

Default value

0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

TCI_DEFAULT_DCU_VAL_LCS_SECURE

The default reset value of the 128-bit DCU signals when LCS=SE and TP Mode=TCI

Type

string

Default value
0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

diagnostics
Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type
int

Default value
0x2

3.10.49 MMC

Generic Multimedia Card. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1026: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for MMC

This model has the following Iris instances:

Table 3-1027: MMC Iris instances

InstanceName	ComponentName
MMC	MMC
MMC.timer	ClockTimerThread
MMC.timer.timer	ClockTimerThread64
MMC.timer.timer.thread	SchedulerThread
MMC.timer.timer.thread_event	SchedulerThreadEvent

MMC contains the following CADI targets:

- [ClockTimerThread](#)
- [ClockTimerThread64](#)
- [MMC](#)
- [SchedulerThread](#)
- [SchedulerThreadEvent](#)

About MMC

This component simulates an SD or SDHC card that is compatible with the *MultiMedia Card Association* (MMCA, <https://www.jedec.org>) specification version 3.31. The parameters permit configuration of a number of attributes reflected in the CID and CSD registers. You can customize the component further by modifying the supplied MMC model source code directly.

When paired with a PL180_MCI component, the MMC device model provides emulation of a flexible, persistent storage mechanism.

The MMC component uses a file on the host PC to simulate the storage device. The size of this backing store file determines the reported size of the MMC device. As small sections of this file are paged in by the model, large filesystems can be modeled while making efficient use of host PC memory. The backing store file can contain a partition table and filesystems such as FAT or EXT2.

The image file is a direct bit copy of the contents of an SD card. If the image file that the `p_mmc_file` parameter refers to does not exist, the component behaves as if the card is absent. If the image file is read-only, then the component behaves as if the card is read-only.



Operating system boots often attempt to write to the boot filesystem. They might not work properly if the boot filesystem is on a read-only card.

The MMC component does not model card insertion or removal. It models the card having already been inserted at system instantiation time.

You can configure the MMC component to behave as an SDHC card by setting the `card_type` parameter to `SDHC`. SDHC mode is a model-specific extension, and is not supported by PL180 hardware. It supports filesystems that are larger than 2GB.

The component supports these commands:

- `MMC_GO_IDLE_STATE.`
- `MMC_SEND_OP_COND.`
- `MMC_ALL_SEND_CID.`
- `MMC_SET_RELATIVE_ADDR.`
- `MMC_SET_DSR.`
- `MMC_SELDESL_CARD.`
- `MMC_SEND_CSD.`
- `MMC_SEND_CID.`
- `MMC_STOP_TRANSMISSION.`
- `MMC_SEND_STATUS.`
- `MMC_GO_INACTIVE_STATE.`
- `MMC_READ_SINGLE_BLOCK.`

- `MMC_READ_MULTIPLE_BLOCK`.
- `MMC_SET_BLOCK_COUNT`.
- `MMC_WRITE_BLOCK`.
- `MMC_WRITE_MULTIPLE_BLOCK`.
- `MMC_SEND_EXT_CSD`. This command is supported in SDHC mode only.

The block length is 512 bytes. SimGen reports attempts to change it as errors.

The component supports these erase commands (Class 5), but they have no effect on the disk backing storage:

- `MMC_ERASE_GROUP_START`.
- `MMC_ERASE_GROUP_END`.
- `MMC_ERASE`.

The component does not support these commands:

- `MMC_BUSTEST_R`.
- `MMC_BUSTEST_W`.

The component does not support stream read and write commands (Classes 1 and 3):

- `MMC_READ_DAT_UNTIL_STOP`.
- `MMC_WRITE_DAT_UNTIL_STOP`.
- `MMC_PROGRAM_CID`.
- `MMC_PROGRAM_CSD`.

The component does not support block oriented write protection commands (Class 6):

- `MMC_SET_WRITE_PROT`.
- `MMC_CLR_WRITE_PROT`.
- `MMC_SEND_WRITE_PROT`.

The component does not support lock card commands (Class 7) or application-specific commands (Class 8):

- `MMC_LOCK_UNLOCK`.
- `MMC_APP_CMD`.
- `MMC_GEN_CMD`.

The component does not support I/O mode commands (Class 9):

- `MMC_FAST_IO`.
- `MMC_GO_IRQ_STATE`.

The component does not support reserved commands. Using a reserved command sets the MMC `ST_ER_B_ILLEGAL_COMMAND` bit in the status register of the card. Read this with the `MMC_SEND_STATUS` command.

Use the `p_diagnostics` parameter to select the level of diagnostic output, to help to debug device driver and controller-to-card protocol issues. It supports the following levels:

Level 0

None.

Level 1

Warnings about attempting to change read-only settings.

Level 2

Trace of command calls.

Level 3

Information about every step in the MMC_Protocol interaction.

Level 4

Hex dump of every block sent or received.

The registers are not memory mapped. Instead, you access them using relevant MMC commands. The MMC component model makes the registers available through a CADI interface. Modification of these registers through CADI is not recommended, but not prohibited. For example, modifying the card ID (CID) registers can be useful when experimenting with drivers, but direct modification of the `STATUS_REG` register is likely to put the card model into an indeterminate state.

For a full definition of MMC registers, see the MMCA System Summary documentation. Device-specific register information can also be obtained from MMC vendors.

Table 3-1028: MMC registers

Name	CADI register number	Description
OCR_REG	0x000	Operating conditions register
CID_REG0	0x004	Card ID bits 127:96
CID_REG1	0x005	Card ID bits 95:64
CID_REG2	0x006	Card ID bits 63:32
CID_REG3	0x007	Card ID bits 31:0
CSD_REG0	0x008	Card specific data bits 127:96
CSD_REG1	0x009	Card specific data bits 95:64
CSD_REG2	0x00a	Card specific data bits 63:32
CSD_REG3	0x00b	Card specific data bit 31:0
RCA_REG	0x00c	Relative card address register
DSR_REG	0x00d	Driver stage register
BLOCKLEN_REG	0x00e	Block length
STATUS_REG	0x00f	Card status
BLOCK_COUNT_REG	0x010	Block count

Ports for MMC

Table 3-1029: Ports

Name	Protocol	Type	Description
card_present	StateSignal	Master	Used to signal whether an MMC image is loaded. It is set if an image is loaded, and is clear if no image is loaded.
clk_in	ClockSignal	Slave	Input clock signal used to drive our 'bus'.
mmc	MMC_Protocol	Slave	The MMC slave port.

Parameters for MMC

- card_type**

Card type ('SD' or 'SDHC')

Type

string

Default value

SDHC
- diagnostics**

Diagnostics level

Type

int

Default value

0x0
- force_sector_addressing**

Use sector addressing even on small cards

Type

bool

Default value

0x0
- p_OEMid**

Card ID OEM ID

Type

int

Default value

0x0
- p_fast_access**

Don't simulate MMC block access delays

Type

bool

Default value

0x1

p_manid

Card ID Manufacturer ID

Type

int

Default value

0x2

p_max_block_count

Default maximum block count reg. Default 0x80

Type

int

Default value

0x80

p_mmc_file

MMCard filename

Type

string

Default value

mmc.dat

p_prodName

Card ID Product Name (6 chars)

Type

string

Default value

ARMmmc

p_prodRev

Card ID Product Revision

Type

int

Default value

0x1

p_sernum

Card Serial Number

Type

int

Default value

0xca4d0001

support_unpadded_images

Support images that are not a multiple of 512k by padding them to the needed size (SDHC cards only)

Type

bool

Default value

0x0

3.10.50 MMU_400

MMU-400 component. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1030: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for MMU_400

This model has the following Iris instances:

Table 3-1031: MMU_400 Iris instances

InstanceName	ComponentName
MMU_400	MMU_400
MMU_400.mmu	MMU_400_BASE
MMU_400.mmu.apb3_control_ns_slv	PVBusSlave
MMU_400.mmu.apb3_control_s_slv	PVBusSlave
MMU_400.mmu.apb4_control_slv	PVBusSlave
MMU_400.mmu.mapper	PVBusMapper
MMU_400.mmu.ptw_dvm_receiver	PVBusMapper
MMU_400.mmu.ptw_master	PVBusMaster
MMU_400.mmu.pvbus_master	PVBusMaster

This model has the following MTI trace components:

Table 3-1032: MMU_400 MTI instances

InstanceName	ComponentName
MMU_400.mmu	MMU_400_BASE
MMU_400.mmu.apb3_control_ns_slv	PVBusSlave
MMU_400.mmu.apb3_control_s_slv	PVBusSlave
MMU_400.mmu.apb4_control_slv	PVBusSlave
MMU_400.mmu.mapper	PVBusMapper
MMU_400.mmu.ptw_dvm_receiver	PVBusMapper
MMU_400.mmu.ptw_master	PVBusMaster
MMU_400.mmu.pvbus_master	PVBusMaster

MMU_400 contains the following CADI targets:

- MMU_400
- MMU_400_BASE

About MMU_400

Set the `use_label_mapping` parameter to `true` if your upstream devices have labels in the top 16 bits of the transaction MasterID.



The model does not have a concept of AXI-ID, but a transaction can have a MasterID set on it.

Label your upstream components 0...*n* so that the parameters of this component can map those integers to StreamID and SSD_Index.

Set `use_label_mapping` to `false` if the StreamID is encoded in the top 16 bits of the MasterID and the bottom 16 bits encode either the SSD_Index or the SSD state directly, depending on `use_ssd_determination_table`. Typically in hardware, a device emits different AXI-IDs, depending on what it is doing. In the model, MasterIDs are usually not diverse and a device might only emit one MasterID.

If `use_ssd_determination_table` is `true`, the bottom 16 bits of the MasterID encode the SSD_Index. They must be $< 2^{\text{ssd_index_width}}$. If it is `false`, they encode the SSD state directly (zero is Secure and nonzero is Non-secure).

This component models all architectural registers that are specified in the Technical Reference Manual (TRM), except that it does not model any of the performance registers, and has the following limitations:

- MMU-400 does not have an SMMU_STLBGSTATUS register because the Secure side is a nominal pass-through. MMU-400 only has stage 2 support and you cannot use stage 2 on the Secure side.

- The SMMU_NSACR is an alias of the Non-secure SMMU_ACR. This component models SMMU_ACR as RAZ/WI.
- The *ACR registers have IMP DEF contents. This component models only the PAGESIZE bit of the SACR, as non-RAZ/WI. It models no other IMP DEF registers.

Ports for MMU_400

Table 3-1033: Ports

Name	Protocol	Type	Description
apb3_control_ns	PVBus	Slave	APBv3 control port for Non-secure access to the register file. If this port is used do not use the APBv4 port.
apb3_control_s	PVBus	Slave	APBv3 control port for Secure access to the register file. If this port is used do not use the APBv4 port.
apb4_control	PVBus	Slave	APBv4 control port for access to the register file. If this port is used do not use the APBv3 ports.
cfg_cttw_in	Signal	Slave	Enables coherent page table walks.
cfgflt_irpt_ns	Signal	Master	Non-secure configuration access fault interrupt. Corresponds to SMMU architectural signal SMMU_NSgCflrpt.
cfgflt_irpt_s	Signal	Master	Secure configuration access fault interrupt. Corresponds to SMMU architectural signal SMMU_gCflrpt.
comb_irpt_ns	Signal	Master	Non-secure combined interrupt.
comb_irpt_s	Signal	Master	Secure combined interrupt.
cxt_irpt_ns	Signal	Master	Non-secure context bank fault.
glblflt_irpt_ns	Signal	Master	Global Non-secure fault interrupt. Corresponds to SMMU architectural signal SMMU_NSglrpt.
glblflt_irpt_s	Signal	Master	Global Secure fault interrupt. Corresponds to SMMU architectural signal SMMU_glrpt.
priv_internals	MMU_400_Internals	Slave	For internal use only, please do not use.
pvbus_m	PVBus	Master	Downstream port of the MMU, where translated transactions emerge.
pvbus_ptw_m	PVBus	Master	Downstream port for page table walks if configured using the ptw_has_separate_port parameter.
pvbus_s	PVBus	Slave	Upstream port of the MMU. Addresses on the port are in VA/IPA.
reset_in	Signal	Slave	Signal to reset the MMU.

Parameters for MMU_400

always_secure_ssd_indices

Non-programmable SSD Indexes that are always secure (e.g. 0, 6, 35-64).

Type

string

Default value

cfg_cttw

Perform coherent page table walks

Type

bool

Default value

0x1

dump_unpredictability_in_user_flags

Override the user flags to encode unpredictable information (validation only)

Type

bool

Default value

0x0

label0_read_ssd

Label0: Read SDD or SSD_Index

Type

int

Default value

0x0

label0_read_stream_id

Label0: Read Stream ID

Type

int

Default value

0x0

label0_write_ssd

Label0: Write SDD or SSD_Index

Type

int

Default value

0x0

label0_write_stream_id

Label0: Write Stream ID

Type

int

Default value

0x0

label10_read_ssd

Label10: Read SDD or SSD_Index

Type

int

Default value

0x0

label10_read_stream_id

Label10: Read Stream ID

Type

int

Default value

0x0

label10_write_ssd

Label10: Write SDD or SSD_Index

Type

int

Default value

0x0

label10_write_stream_id

Label10: Write Stream ID

Type

int

Default value

0x0

label11_read_ssd

Label11: Read SDD or SSD_Index

Type

int

Default value

0x0

label11_read_stream_id

Label11: Read Stream ID

Type

int

Default value

0x0

label11_write_ssd

Label11: Write SDD or SSD_Index

Type

int

Default value

0x0

label11_write_stream_id

Label11: Write Stream ID

Type

int

Default value

0x0

label12_read_ssd

Label12: Read SDD or SSD_Index

Type

int

Default value

0x0

label12_read_stream_id

Label12: Read Stream ID

Type

int

Default value

0x0

label12_write_ssd

Label12: Write SDD or SSD_Index

Type

int

Default value

0x0

label12_write_stream_id

Label12: Write Stream ID

Type

int

Default value

0x0

label13_read_ssd

Label13: Read SDD or SSD_Index

Type

int

Default value

0x0

label13_read_stream_id

Label13: Read Stream ID

Type

int

Default value

0x0

label13_write_ssd

Label13: Write SDD or SSD_Index

Type

int

Default value

0x0

label13_write_stream_id

Label13: Write Stream ID

Type

int

Default value

0x0

label14_read_ssd

Label14: Read SDD or SSD_Index

Type

int

Default value

0x0

label14_read_stream_id

Label14: Read Stream ID

Type

int

Default value

0x0

label14_write_ssd

Label14: Write SDD or SSD_Index

Type

int

Default value

0x0

label14_write_stream_id

Label14: Write Stream ID

Type

int

Default value

0x0

label15_read_ssd

Label15: Read SDD or SSD_Index

Type

int

Default value

0x0

label15_read_stream_id

Label15: Read Stream ID

Type

int

Default value

0x0

label15_write_ssd

Label15: Write SDD or SSD_Index

Type

int

Default value

0x0

label15_write_stream_id

Label15: Write Stream ID

Type

int

Default value

0x0

label16_read_ssd

Label16: Read SDD or SSD_Index

Type

int

Default value

0x0

label16_read_stream_id

Label16: Read Stream ID

Type

int

Default value

0x0

label16_write_ssd

Label16: Write SDD or SSD_Index

Type

int

Default value

0x0

label16_write_stream_id

Label16: Write Stream ID

Type

int

Default value

0x0

label17_read_ssd

Label17: Read SDD or SSD_Index

Type

int

Default value

0x0

label17_read_stream_id

Label17: Read Stream ID

Type

int

Default value

0x0

label17_write_ssd

Label17: Write SDD or SSD_Index

Type

int

Default value

0x0

label17_write_stream_id

Label17: Write Stream ID

Type

int

Default value

0x0

label18_read_ssd

Label18: Read SDD or SSD_Index

Type

int

Default value

0x0

label18_read_stream_id

Label18: Read Stream ID

Type

int

Default value

0x0

label18_write_ssd

Label18: Write SDD or SSD_Index

Type

int

Default value

0x0

label18_write_stream_id

Label18: Write Stream ID

Type

int

Default value

0x0

label19_read_ssd

Label19: Read SDD or SSD_Index

Type

int

Default value

0x0

label19_read_stream_id

Label19: Read Stream ID

Type

int

Default value

0x0

label19_write_ssd

Label19: Write SDD or SSD_Index

Type

int

Default value

0x0

label19_write_stream_id

Label19: Write Stream ID

Type

int

Default value

0x0

label11_read_ssd

Label1: Read SDD or SSD_Index

Type

int

Default value

0x0

label11_read_stream_id

Label1: Read Stream ID

Type

int

Default value

0x0

label1_write_ssd

Label1: Write SDD or SSD_Index

Type

int

Default value

0x0

label1_write_stream_id

Label1: Write Stream ID

Type

int

Default value

0x0

label20_read_ssd

Label20: Read SDD or SSD_Index

Type

int

Default value

0x0

label20_read_stream_id

Label20: Read Stream ID

Type

int

Default value

0x0

label20_write_ssd

Label20: Write SDD or SSD_Index

Type

int

Default value

0x0

label20_write_stream_id

Label20: Write Stream ID

Type

int

Default value

0x0

label21_read_ssd

Label21: Read SDD or SSD_Index

Type

int

Default value

0x0

label21_read_stream_id

Label21: Read Stream ID

Type

int

Default value

0x0

label21_write_ssd

Label21: Write SDD or SSD_Index

Type

int

Default value

0x0

label21_write_stream_id

Label21: Write Stream ID

Type

int

Default value

0x0

label22_read_ssd

Label22: Read SDD or SSD_Index

Type

int

Default value

0x0

label22_read_stream_id

Label22: Read Stream ID

Type

int

Default value

0x0

label122_write_ssd

Label22: Write SDD or SSD_Index

Type

int

Default value

0x0

label122_write_stream_id

Label22: Write Stream ID

Type

int

Default value

0x0

label123_read_ssd

Label23: Read SDD or SSD_Index

Type

int

Default value

0x0

label123_read_stream_id

Label23: Read Stream ID

Type

int

Default value

0x0

label123_write_ssd

Label23: Write SDD or SSD_Index

Type

int

Default value

0x0

label123_write_stream_id

Label23: Write Stream ID

Type

int

Default value

0x0

label24_read_ssd

Label24: Read SDD or SSD_Index

Type

int

Default value

0x0

label24_read_stream_id

Label24: Read Stream ID

Type

int

Default value

0x0

label24_write_ssd

Label24: Write SDD or SSD_Index

Type

int

Default value

0x0

label24_write_stream_id

Label24: Write Stream ID

Type

int

Default value

0x0

label25_read_ssd

Label25: Read SDD or SSD_Index

Type

int

Default value

0x0

label25_read_stream_id

Label25: Read Stream ID

Type

int

Default value

0x0

label25_write_ssd

Label25: Write SDD or SSD_Index

Type

int

Default value

0x0

label25_write_stream_id

Label25: Write Stream ID

Type

int

Default value

0x0

label26_read_ssd

Label26: Read SDD or SSD_Index

Type

int

Default value

0x0

label26_read_stream_id

Label26: Read Stream ID

Type

int

Default value

0x0

label26_write_ssd

Label26: Write SDD or SSD_Index

Type

int

Default value

0x0

label26_write_stream_id

Label26: Write Stream ID

Type

int

Default value

0x0

label127_read_ssd

Label27: Read SDD or SSD_Index

Type

int

Default value

0x0

label127_read_stream_id

Label27: Read Stream ID

Type

int

Default value

0x0

label127_write_ssd

Label27: Write SDD or SSD_Index

Type

int

Default value

0x0

label127_write_stream_id

Label27: Write Stream ID

Type

int

Default value

0x0

label128_read_ssd

Label28: Read SDD or SSD_Index

Type

int

Default value

0x0

label128_read_stream_id

Label28: Read Stream ID

Type

int

Default value

0x0

label28_write_ssd

Label28: Write SDD or SSD_Index

Type

int

Default value

0x0

label28_write_stream_id

Label28: Write Stream ID

Type

int

Default value

0x0

label29_read_ssd

Label29: Read SDD or SSD_Index

Type

int

Default value

0x0

label29_read_stream_id

Label29: Read Stream ID

Type

int

Default value

0x0

label29_write_ssd

Label29: Write SDD or SSD_Index

Type

int

Default value

0x0

label29_write_stream_id

Label29: Write Stream ID

Type

int

Default value

0x0

label12_read_ssd

Label2: Read SDD or SSD_Index

Type

int

Default value

0x0

label12_read_stream_id

Label2: Read Stream ID

Type

int

Default value

0x0

label12_write_ssd

Label2: Write SDD or SSD_Index

Type

int

Default value

0x0

label12_write_stream_id

Label2: Write Stream ID

Type

int

Default value

0x0

label130_read_ssd

Label30: Read SDD or SSD_Index

Type

int

Default value

0x0

label130_read_stream_id

Label30: Read Stream ID

Type

int

Default value

0x0

label30_write_ssd

Label30: Write SDD or SSD_Index

Type

int

Default value

0x0

label30_write_stream_id

Label30: Write Stream ID

Type

int

Default value

0x0

label31_read_ssd

Label31: Read SDD or SSD_Index

Type

int

Default value

0x0

label31_read_stream_id

Label31: Read Stream ID

Type

int

Default value

0x0

label31_write_ssd

Label31: Write SDD or SSD_Index

Type

int

Default value

0x0

label31_write_stream_id

Label31: Write Stream ID

Type

int

Default value

0x0

label13_read_ssd

Label3: Read SDD or SSD_Index

Type

int

Default value

0x0

label13_read_stream_id

Label3: Read Stream ID

Type

int

Default value

0x0

label13_write_ssd

Label3: Write SDD or SSD_Index

Type

int

Default value

0x0

label13_write_stream_id

Label3: Write Stream ID

Type

int

Default value

0x0

label14_read_ssd

Label4: Read SDD or SSD_Index

Type

int

Default value

0x0

label14_read_stream_id

Label4: Read Stream ID

Type

int

Default value

0x0

label4_write_ssd

Label4: Write SDD or SSD_Index

Type

int

Default value

0x0

label4_write_stream_id

Label4: Write Stream ID

Type

int

Default value

0x0

label5_read_ssd

Label5: Read SDD or SSD_Index

Type

int

Default value

0x0

label5_read_stream_id

Label5: Read Stream ID

Type

int

Default value

0x0

label5_write_ssd

Label5: Write SDD or SSD_Index

Type

int

Default value

0x0

label5_write_stream_id

Label5: Write Stream ID

Type

int

Default value

0x0

label6_read_ssd

Label6: Read SDD or SSD_Index

Type

int

Default value

0x0

label6_read_stream_id

Label6: Read Stream ID

Type

int

Default value

0x0

label6_write_ssd

Label6: Write SDD or SSD_Index

Type

int

Default value

0x0

label6_write_stream_id

Label6: Write Stream ID

Type

int

Default value

0x0

label7_read_ssd

Label7: Read SDD or SSD_Index

Type

int

Default value

0x0

label7_read_stream_id

Label7: Read Stream ID

Type

int

Default value

0x0

label7_write_ssd

Label7: Write SDD or SSD_Index

Type

int

Default value

0x0

label7_write_stream_id

Label7: Write Stream ID

Type

int

Default value

0x0

label8_read_ssd

Label8: Read SDD or SSD_Index

Type

int

Default value

0x0

label8_read_stream_id

Label8: Read Stream ID

Type

int

Default value

0x0

label8_write_ssd

Label8: Write SDD or SSD_Index

Type

int

Default value

0x0

label8_write_stream_id

Label8: Write Stream ID

Type

int

Default value

0x0

label9_read_ssd

Label9: Read SDD or SSD_Index

Type

int

Default value

0x0

label9_read_stream_id

Label9: Read Stream ID

Type

int

Default value

0x0

label9_write_ssd

Label9: Write SDD or SSD_Index

Type

int

Default value

0x0

label9_write_stream_id

Label9: Write Stream ID

Type

int

Default value

0x0

mmu.seed

Seed for SMMU

Type

int

Default value

0x12345678

number_of_contexts

Number of context banks

Type

int

Default value

0x8

number_of_smrs

Number of stream match registers.

Type

int

Default value

0x20

percent_tlbstatus_commits

Percentage of times that a poll of TLBSTATUS will commit the TLBI commands

Type

int

Default value

0xa

prefetch_only_requests

Handle prefetch-only requests by:- 0 -- deny them 1 -- use debug table walks and TLB entries 2 -- treat them as normal transactions (dangerous)

Type

int

Default value

0x0

programmable_non_secure_by_default_ssd_indices

Programmable SSD Indexes that are by default non-secure (e.g. 0, 6, 35-84).

Type

string

Default value**programmable_secure_by_default_ssd_indices**

Programmable SSD Indexes that are by default secure (e.g. 0, 6, 35-84).

Type

string

Default value**ptw_has_separate_port**

Page Table Walks use pvbus_ptw_m

Type

bool

Default value

0x1

pvbus_m_is_ace_lite

Is pvbus_m (the downstream port that translated transaction exit) ACE-Lite

Type

bool

Default value

0x1

pvbus_ptw_m_is_ace_lite

Is pvbus_ptw_m (the downstream port that is used for walks if ptw_has_separate_port is true) ACE-Lite

Type

bool

Default value

0x1

stream_id_width

StreamID bit width

Type

int

Default value

0x6

tlb_depth

TLB Depth (0 means 10000). The model will perform best with more TLB entries.

Type

int

Default value

0x40

use_label_mapping

Use label mapping

Type

bool

Default value

0x1

use_ssd_determination_table

Use SSD Determination Table

Type

bool

Default value

0x1

3.10.51 MMU_400_BASE

MMU-400 base component. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1034: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for MMU_400_BASE

This model has the following Iris instances:

Table 3-1035: MMU_400_BASE Iris instances

InstanceName	ComponentName
MMU_400_BASE	MMU_400_BASE
MMU_400_BASE.apb3_control_ns_slv	PVBusSlave
MMU_400_BASE.apb3_control_s_slv	PVBusSlave
MMU_400_BASE.apb4_control_slv	PVBusSlave
MMU_400_BASE.mapper	PVBusMapper
MMU_400_BASE.ptw_dvm_receiver	PVBusMapper
MMU_400_BASE.ptw_master	PVBusMaster
MMU_400_BASE.pvbus_master	PVBusMaster

This model has the following MTI trace components:

Table 3-1036: MMU_400_BASE MTI instances

InstanceName	ComponentName
MMU_400_BASE	MMU_400_BASE
MMU_400_BASE.apb3_control_ns_slv	PVBusSlave
MMU_400_BASE.apb3_control_s_slv	PVBusSlave
MMU_400_BASE.apb4_control_slv	PVBusSlave
MMU_400_BASE.mapper	PVBusMapper
MMU_400_BASE.ptw_dvm_receiver	PVBusMapper
MMU_400_BASE.ptw_master	PVBusMaster
MMU_400_BASE.pvbus_master	PVBusMaster

MMU_400_BASE contains the following CADI targets:

- MMU_400_BASE

Ports for MMU_400_BASE

Table 3-1037: Ports

Name	Protocol	Type	Description
apb3_control_ns	PVBus	Slave	If the device has been configured with APB3 control ports then this is used to address the register file with non-secure accesses. If this is the case then the apb4_control port should not be used.
apb3_control_s	PVBus	Slave	If the device has been configured with APB3 control ports then this is used to address the register file with secure accesses. If this is the case then the apb4_control port should not be used.
apb4_control	PVBus	Slave	If the device has been configured with APB4 control ports then this port is used -- it carries the security world with the transaction itself. If this is the case then the apb3_control_s and apb3_control_ns should not be used.
cfg_cttw_in	Signal	Slave	The SoC supports coherent page walks, this is meant to be sampled at reset. However, in practice the model has to prevent the race condition between cfg_cttw being asserted at the same 'cycle' as negedge reset. Thus we actually only sample the signal on the first transaction to the SMMU or the first transition on this signal after reset. Thus in the model, we require that cfg_cttw be held for at least this period of time.
cfgflt_irpt_ns	Signal	Master	Non-secure Configuration Access Fault Interrupt In the SMMU Architecture this is called SMMU_NSgCflrpt.
cfgflt_irpt_s	Signal	Master	Secure Configuration Access Fault Interrupt In the SMMU Architecture this is called SMMU_gCflrpt.
comb_irpt_ns	Signal	Master	"Non-secure combined interrupt" (cfgflt_irpt_ns glblflt_irpt_ns cxt_irpt_ns)?
comb_irpt_s	Signal	Master	"Secure combined interrupt"
cxt_irpt_ns	Signal	Master	Non-secure context bank fault NOTE that there is only one context bank fault, despite there being potentially 8 contexts. As we are HW stage 2 only then we can't have any banks configured as secure (well if we do then we generate a global fault).
glblflt_irpt_ns	Signal	Master	Global non-secure Fault Interrupt In the SMMU Architecture this is called SMMU_NSglrpt.
glblflt_irpt_s	Signal	Master	Global secure Fault Interrupt In the SMMU Architecture this is called SMMU_glrpt.
identify	MMU_400_BASE_IDENTIFY	Master	This port is a special model port that is used to take a transaction and map it to an SSD/SSD_Index and StreamID.
priv_internals	MMU_400_Internals	Slave	For internal use only, please do not use.
pvbus_m	PVBus	Master	This downstream port is where the translated accesses from pvbus_s emerge. If page walks are configured to come out of this port, then they will come out with the with the same attributes as described for pvbus_ptw_m.

Name	Protocol	Type	Description
pvbus_ptw_m	PVBus	Master	This downstream port is where page table walk accesses come from. This is only used if configured to use a separate page table walk port. The MMU-400 will only obey DVM messages if configured to use this port. The page walks come out of this port with the following master_id and user_flags. master_id : 0xFFFFFFFF The user flags : user_flags[7:0] stage 1 context_id (or 0xFF if stage2 only) user_flags[15:8] stage 2 context_id (or 0xFF if stage 1 with stage 2 bypass) user_flags[18:16] stage 1 level user_flags[21:19] stage 2 level user_flags[23:22] stage 1 descriptor encoding (0=v7s, 1=v7l, 2=v8l, 3=none) user_flags[25:24] stage 2 descriptor encoding (0=v7s, 1=v7l, 2=v8l, 3=none) user_flags[31,30] adomain of the transaction NOTE that if the walk is being done for a stage 1 page walk descriptor fetch then the stage 1 level field will indicate that level. If the walk is being done for a stage 2 descriptor fetch, then the stage 2 level field will show that level. If the context-id for a stage is not valid (0xFF) then the 'level' information is 0x7.
pvbus_s	PVBus	Slave	This port is the upstream port of the device, addresses on the port are in the VA/IPA
reset_in	Signal	Slave	The reset pin.

Parameters for MMU_400_BASE

always_secure_ssd_indices

Non-programmable SSD Indexes that are always secure (e.g. 0, 6, 35-64).

Type

string

Default value

cfg_cttw

Perform coherent page table walks

Type

bool

Default value

0x1

dump_unpredictability_in_user_flags

Override the user flags to encode unpredictable information (validation only)

Type

bool

Default value

0x0

number_of_contexts

Number of context banks

Type

int

Default value

0x8

number_of_smrs

Number of stream match registers.

Type

int

Default value

0x10

percent_tlbstatus_commits

Percentage of times that a poll of TLBSTATUS will commit the TLBI commands

Type

int

Default value

0xa

prefetch_only_requests

Handle prefetch-only requests by:- 0 -- deny them 1 -- use debug table walks and TLB entries 2 -- treat them as normal transactions (dangerous)

Type

int

Default value

0x0

programmable_non_secure_by_default_ssd_indices

Programmable SSD Indexes that are by default non-secure (e.g. 0, 6, 35-84).

Type

string

Default value**programmable_secure_by_default_ssd_indices**

Programmable SSD Indexes that are by default secure (e.g. 0, 6, 35-84).

Type

string

Default value**ptw_has_separate_port**

Page Table Walks use pvbus_ptw_m

Type

bool

Default value

0x1

pvbus_m_is_ace_lite

Is pvbus_m (the downstream port that translated transaction exit) ACE-Lite

Type

bool

Default value

0x1

pvbus_ptw_m_is_ace_lite

Is pvbus_ptw_m (the downstream port that is used for walks if ptw_has_separate_port is true) ACE-Lite

Type

bool

Default value

0x1

seed

Seed for SMMU

Type

int

Default value

0x12345678

stream_id_width

StreamID bit width

Type

int

Default value

0x6

tlb_depth

TLB Depth (0 means 10000). The model will perform best with more TLB entries.

Type

int

Default value

0x40

use_ssd_determination_table

Use SSD Determination Table

Type

bool

Default value

0x1

3.10.52 MMU_500

MMU-500 component. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1038: IP revisions support

Revision	Quality level
r2p4	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for MMU_500

This model has the following Iris instances:

Table 3-1039: MMU_500 Iris instances

InstanceName	ComponentName
MMU_500	MMU_500
MMU_500.mmu	MMU_500_BASE
MMU_500.mmu.mapper0	PVBusMapper
MMU_500.mmu.mapper1	PVBusMapper
MMU_500.mmu.mapper10	PVBusMapper
MMU_500.mmu.mapper11	PVBusMapper
MMU_500.mmu.mapper12	PVBusMapper
MMU_500.mmu.mapper13	PVBusMapper
MMU_500.mmu.mapper14	PVBusMapper
MMU_500.mmu.mapper15	PVBusMapper
MMU_500.mmu.mapper16	PVBusMapper
MMU_500.mmu.mapper17	PVBusMapper
MMU_500.mmu.mapper18	PVBusMapper
MMU_500.mmu.mapper19	PVBusMapper
MMU_500.mmu.mapper2	PVBusMapper
MMU_500.mmu.mapper20	PVBusMapper
MMU_500.mmu.mapper21	PVBusMapper
MMU_500.mmu.mapper22	PVBusMapper
MMU_500.mmu.mapper23	PVBusMapper
MMU_500.mmu.mapper24	PVBusMapper

InstanceName	ComponentName
MMU_500.mmu.mapper25	PVBusMapper
MMU_500.mmu.mapper26	PVBusMapper
MMU_500.mmu.mapper27	PVBusMapper
MMU_500.mmu.mapper28	PVBusMapper
MMU_500.mmu.mapper29	PVBusMapper
MMU_500.mmu.mapper3	PVBusMapper
MMU_500.mmu.mapper30	PVBusMapper
MMU_500.mmu.mapper31	PVBusMapper
MMU_500.mmu.mapper4	PVBusMapper
MMU_500.mmu.mapper5	PVBusMapper
MMU_500.mmu.mapper6	PVBusMapper
MMU_500.mmu.mapper7	PVBusMapper
MMU_500.mmu.mapper8	PVBusMapper
MMU_500.mmu.mapper9	PVBusMapper
MMU_500.mmu.ptw_dvm_receiver	PVBusMapper
MMU_500.mmu.ptw_master	PVBusMaster
MMU_500.mmu.pvbus_control_s_slv	PVBusSlave
MMU_500.mmu.pvbus_master	PVBusMaster
MMU_500.mmu.pvbusmaster0	PVBusMaster
MMU_500.mmu.pvbusmaster1	PVBusMaster
MMU_500.mmu.pvbusmaster10	PVBusMaster
MMU_500.mmu.pvbusmaster11	PVBusMaster
MMU_500.mmu.pvbusmaster12	PVBusMaster
MMU_500.mmu.pvbusmaster13	PVBusMaster
MMU_500.mmu.pvbusmaster14	PVBusMaster
MMU_500.mmu.pvbusmaster15	PVBusMaster
MMU_500.mmu.pvbusmaster16	PVBusMaster
MMU_500.mmu.pvbusmaster17	PVBusMaster
MMU_500.mmu.pvbusmaster18	PVBusMaster
MMU_500.mmu.pvbusmaster19	PVBusMaster
MMU_500.mmu.pvbusmaster2	PVBusMaster
MMU_500.mmu.pvbusmaster20	PVBusMaster
MMU_500.mmu.pvbusmaster21	PVBusMaster
MMU_500.mmu.pvbusmaster22	PVBusMaster
MMU_500.mmu.pvbusmaster23	PVBusMaster
MMU_500.mmu.pvbusmaster24	PVBusMaster
MMU_500.mmu.pvbusmaster25	PVBusMaster
MMU_500.mmu.pvbusmaster26	PVBusMaster
MMU_500.mmu.pvbusmaster27	PVBusMaster

InstanceName	ComponentName
MMU_500.mmu.pvbusmaster28	PVBusMaster
MMU_500.mmu.pvbusmaster29	PVBusMaster
MMU_500.mmu.pvbusmaster3	PVBusMaster
MMU_500.mmu.pvbusmaster30	PVBusMaster
MMU_500.mmu.pvbusmaster31	PVBusMaster
MMU_500.mmu.pvbusmaster4	PVBusMaster
MMU_500.mmu.pvbusmaster5	PVBusMaster
MMU_500.mmu.pvbusmaster6	PVBusMaster
MMU_500.mmu.pvbusmaster7	PVBusMaster
MMU_500.mmu.pvbusmaster8	PVBusMaster
MMU_500.mmu.pvbusmaster9	PVBusMaster
MMU_500.mmu.pvbusslave0	PVBusSlave
MMU_500.mmu.pvbusslave1	PVBusSlave
MMU_500.mmu.pvbusslave10	PVBusSlave
MMU_500.mmu.pvbusslave11	PVBusSlave
MMU_500.mmu.pvbusslave12	PVBusSlave
MMU_500.mmu.pvbusslave13	PVBusSlave
MMU_500.mmu.pvbusslave14	PVBusSlave
MMU_500.mmu.pvbusslave15	PVBusSlave
MMU_500.mmu.pvbusslave16	PVBusSlave
MMU_500.mmu.pvbusslave17	PVBusSlave
MMU_500.mmu.pvbusslave18	PVBusSlave
MMU_500.mmu.pvbusslave19	PVBusSlave
MMU_500.mmu.pvbusslave2	PVBusSlave
MMU_500.mmu.pvbusslave20	PVBusSlave
MMU_500.mmu.pvbusslave21	PVBusSlave
MMU_500.mmu.pvbusslave22	PVBusSlave
MMU_500.mmu.pvbusslave23	PVBusSlave
MMU_500.mmu.pvbusslave24	PVBusSlave
MMU_500.mmu.pvbusslave25	PVBusSlave
MMU_500.mmu.pvbusslave26	PVBusSlave
MMU_500.mmu.pvbusslave27	PVBusSlave
MMU_500.mmu.pvbusslave28	PVBusSlave
MMU_500.mmu.pvbusslave29	PVBusSlave
MMU_500.mmu.pvbusslave3	PVBusSlave
MMU_500.mmu.pvbusslave30	PVBusSlave
MMU_500.mmu.pvbusslave31	PVBusSlave
MMU_500.mmu.pvbusslave4	PVBusSlave
MMU_500.mmu.pvbusslave5	PVBusSlave

InstanceName	ComponentName
MMU_500.mmu.pvbusslave6	PVBusSlave
MMU_500.mmu.pvbusslave7	PVBusSlave
MMU_500.mmu.pvbusslave8	PVBusSlave
MMU_500.mmu.pvbusslave9	PVBusSlave

This model has the following MTI trace components:

Table 3-1040: MMU_500 MTI instances

InstanceName	ComponentName
MMU_500.mmu	MMU_500_BASE
MMU_500.mmu.mapper0	PVBusMapper
MMU_500.mmu.mapper1	PVBusMapper
MMU_500.mmu.mapper10	PVBusMapper
MMU_500.mmu.mapper11	PVBusMapper
MMU_500.mmu.mapper12	PVBusMapper
MMU_500.mmu.mapper13	PVBusMapper
MMU_500.mmu.mapper14	PVBusMapper
MMU_500.mmu.mapper15	PVBusMapper
MMU_500.mmu.mapper16	PVBusMapper
MMU_500.mmu.mapper17	PVBusMapper
MMU_500.mmu.mapper18	PVBusMapper
MMU_500.mmu.mapper19	PVBusMapper
MMU_500.mmu.mapper2	PVBusMapper
MMU_500.mmu.mapper20	PVBusMapper
MMU_500.mmu.mapper21	PVBusMapper
MMU_500.mmu.mapper22	PVBusMapper
MMU_500.mmu.mapper23	PVBusMapper
MMU_500.mmu.mapper24	PVBusMapper
MMU_500.mmu.mapper25	PVBusMapper
MMU_500.mmu.mapper26	PVBusMapper
MMU_500.mmu.mapper27	PVBusMapper
MMU_500.mmu.mapper28	PVBusMapper
MMU_500.mmu.mapper29	PVBusMapper
MMU_500.mmu.mapper3	PVBusMapper
MMU_500.mmu.mapper30	PVBusMapper
MMU_500.mmu.mapper31	PVBusMapper
MMU_500.mmu.mapper4	PVBusMapper
MMU_500.mmu.mapper5	PVBusMapper
MMU_500.mmu.mapper6	PVBusMapper
MMU_500.mmu.mapper7	PVBusMapper

InstanceName	ComponentName
MMU_500.mmu.mapper8	PVBusMapper
MMU_500.mmu.mapper9	PVBusMapper
MMU_500.mmu.ptw_dvm_receiver	PVBusMapper
MMU_500.mmu.ptw_master	PVBusMaster
MMU_500.mmu.pvbus_control_s_slv	PVBusSlave
MMU_500.mmu.pvbus_master	PVBusMaster
MMU_500.mmu.pvbusmaster0	PVBusMaster
MMU_500.mmu.pvbusmaster1	PVBusMaster
MMU_500.mmu.pvbusmaster10	PVBusMaster
MMU_500.mmu.pvbusmaster11	PVBusMaster
MMU_500.mmu.pvbusmaster12	PVBusMaster
MMU_500.mmu.pvbusmaster13	PVBusMaster
MMU_500.mmu.pvbusmaster14	PVBusMaster
MMU_500.mmu.pvbusmaster15	PVBusMaster
MMU_500.mmu.pvbusmaster16	PVBusMaster
MMU_500.mmu.pvbusmaster17	PVBusMaster
MMU_500.mmu.pvbusmaster18	PVBusMaster
MMU_500.mmu.pvbusmaster19	PVBusMaster
MMU_500.mmu.pvbusmaster2	PVBusMaster
MMU_500.mmu.pvbusmaster20	PVBusMaster
MMU_500.mmu.pvbusmaster21	PVBusMaster
MMU_500.mmu.pvbusmaster22	PVBusMaster
MMU_500.mmu.pvbusmaster23	PVBusMaster
MMU_500.mmu.pvbusmaster24	PVBusMaster
MMU_500.mmu.pvbusmaster25	PVBusMaster
MMU_500.mmu.pvbusmaster26	PVBusMaster
MMU_500.mmu.pvbusmaster27	PVBusMaster
MMU_500.mmu.pvbusmaster28	PVBusMaster
MMU_500.mmu.pvbusmaster29	PVBusMaster
MMU_500.mmu.pvbusmaster3	PVBusMaster
MMU_500.mmu.pvbusmaster30	PVBusMaster
MMU_500.mmu.pvbusmaster31	PVBusMaster
MMU_500.mmu.pvbusmaster4	PVBusMaster
MMU_500.mmu.pvbusmaster5	PVBusMaster
MMU_500.mmu.pvbusmaster6	PVBusMaster
MMU_500.mmu.pvbusmaster7	PVBusMaster
MMU_500.mmu.pvbusmaster8	PVBusMaster
MMU_500.mmu.pvbusmaster9	PVBusMaster
MMU_500.mmu.pvbusslave0	PVBusSlave

InstanceName	ComponentName
MMU_500.mmu.pvbusslave1	PVBusSlave
MMU_500.mmu.pvbusslave10	PVBusSlave
MMU_500.mmu.pvbusslave11	PVBusSlave
MMU_500.mmu.pvbusslave12	PVBusSlave
MMU_500.mmu.pvbusslave13	PVBusSlave
MMU_500.mmu.pvbusslave14	PVBusSlave
MMU_500.mmu.pvbusslave15	PVBusSlave
MMU_500.mmu.pvbusslave16	PVBusSlave
MMU_500.mmu.pvbusslave17	PVBusSlave
MMU_500.mmu.pvbusslave18	PVBusSlave
MMU_500.mmu.pvbusslave19	PVBusSlave
MMU_500.mmu.pvbusslave2	PVBusSlave
MMU_500.mmu.pvbusslave20	PVBusSlave
MMU_500.mmu.pvbusslave21	PVBusSlave
MMU_500.mmu.pvbusslave22	PVBusSlave
MMU_500.mmu.pvbusslave23	PVBusSlave
MMU_500.mmu.pvbusslave24	PVBusSlave
MMU_500.mmu.pvbusslave25	PVBusSlave
MMU_500.mmu.pvbusslave26	PVBusSlave
MMU_500.mmu.pvbusslave27	PVBusSlave
MMU_500.mmu.pvbusslave28	PVBusSlave
MMU_500.mmu.pvbusslave29	PVBusSlave
MMU_500.mmu.pvbusslave3	PVBusSlave
MMU_500.mmu.pvbusslave30	PVBusSlave
MMU_500.mmu.pvbusslave31	PVBusSlave
MMU_500.mmu.pvbusslave4	PVBusSlave
MMU_500.mmu.pvbusslave5	PVBusSlave
MMU_500.mmu.pvbusslave6	PVBusSlave
MMU_500.mmu.pvbusslave7	PVBusSlave
MMU_500.mmu.pvbusslave8	PVBusSlave
MMU_500.mmu.pvbusslave9	PVBusSlave

MMU_500 contains the following CADI targets:

- MMU_500
- MMU_500_BASE

About MMU_500

This is a model of a basic MMU-500. Set the version using the `version` parameter.

You cannot arbitrarily configure how you derive StreamIDs and SSD_Indexes from the transaction attributes.

This component has two label modes which you select using the parameter `use_label_mapping`:

- Set `use_label_mapping` to `true` if your upstream devices have labels in the top 16 bits of the transaction MasterID.



The model does not have a concept of AXI-ID, but a transaction can have a MasterID set on it.

Label your upstream components 0...*N* so that the parameters of this component can map those integers to StreamID and SSD_Index.

- Set `use_label_mapping` to `false` if the StreamID is encoded in the top 16 bits of the MasterID and the bottom 16 bits encode either the SSD_Index or the SSD state directly, depending on `use_ssd_determination_table`:
 - If `use_ssd_determination_table` is `true`, the bottom 16 bits of the MasterID encode the SSD_Index. They must be $< 2^{\text{ssd_index_width}}$.
 - If `use_ssd_determination_table` is `false`, the bottom 16 bits of the MasterID encode the SSD state directly, where zero is Secure and nonzero is Non-secure.

Typically in hardware, a device emits different AXI-IDs, depending on what it is doing. In the model, MasterIDs are usually not diverse and a device might only emit one MasterID.

This component models the registers as follows:

- It models all architectural registers that the Technical Reference Manual (TRM) specifies, except that it does not model any of the performance registers.
- Unlike the MMU-400, MMU-500 does have an SMMU_STLBGSTATUS register because it has stage 1 and stage 2 support.
- The SMMU_NSACR is an alias of the Non-secure SMMU_ACR. This component models SMMU_ACR as RAZ/WI.
- The *ACR registers have IMP DEF contents. This component models only the PAGESIZE bit of the SACR, as non-RAZ/WI. It models no other IMP DEF registers.

Ports for MMU_500

Table 3-1041: Ports

Name	Protocol	Type	Description
<code>cfg_cttw_in</code>	Signal	Slave	Enables coherent page table walks.
<code>comb_irpt_ns</code>	Signal	Master	Non-secure combined interrupt.
<code>comb_irpt_s</code>	Signal	Master	Secure combined interrupt.
<code>cxt_irpt[128]</code>	Signal	Master	Context interrupt.
<code>glblflt_irpt_ns</code>	Signal	Master	Global Non-secure fault interrupt.

Name	Protocol	Type	Description
glbl_flt_irpt_s	Signal	Master	Global Secure fault interrupt.
priv_internals	MMU_500_Internals	Slave	For internal use only, please do not use.
pvbus_control_s	PVBus	Slave	Provides memory-mapped read write access to the control registers of the module.
pvbus_m[32]	PVBus	Master	For all memory accesses. One for each Translation Buffer Unit (TBU).
pvbus_ptw_m	PVBus	Master	If ptw_has_separate_port is true, use for page table walks.
pvbus_s[32]	PVBus	Slave	For transactions from PVBus master/decoder. One for each TBU.
reset_in	Signal	Slave	Reset signal.

Parameters for MMU_500

always_secure_ssd_indices

Non-programmable SSD Indexes that are always secure (e.g. 0, 6, 35-64).

Type

string

Default value

cfg_cttw

Perform coherent page table walks

Type

bool

Default value

0x1

dump_unpredictability_in_user_flags

Override the user flags to encode unpredictable information (validation only)

Type

bool

Default value

0x0

label0_read_ssd

Label0: Read SDD or SSD_Index

Type

int

Default value

0x0

label0_read_stream_id

Label0: Read Stream ID

Type

int

Default value

0x0

label10_write_ssd

Label10: Write SDD or SSD_Index

Type

int

Default value

0x0

label10_write_stream_id

Label10: Write Stream ID

Type

int

Default value

0x0

label110_read_ssd

Label110: Read SDD or SSD_Index

Type

int

Default value

0x0

label110_read_stream_id

Label110: Read Stream ID

Type

int

Default value

0x0

label110_write_ssd

Label110: Write SDD or SSD_Index

Type

int

Default value

0x0

label10_write_stream_id

Label10: Write Stream ID

Type

int

Default value

0x0

label11_read_ssd

Label11: Read SDD or SSD_Index

Type

int

Default value

0x0

label11_read_stream_id

Label11: Read Stream ID

Type

int

Default value

0x0

label11_write_ssd

Label11: Write SDD or SSD_Index

Type

int

Default value

0x0

label11_write_stream_id

Label11: Write Stream ID

Type

int

Default value

0x0

label12_read_ssd

Label12: Read SDD or SSD_Index

Type

int

Default value

0x0

label12_read_stream_id

Label12: Read Stream ID

Type

int

Default value

0x0

label12_write_ssd

Label12: Write SDD or SSD_Index

Type

int

Default value

0x0

label12_write_stream_id

Label12: Write Stream ID

Type

int

Default value

0x0

label13_read_ssd

Label13: Read SDD or SSD_Index

Type

int

Default value

0x0

label13_read_stream_id

Label13: Read Stream ID

Type

int

Default value

0x0

label13_write_ssd

Label13: Write SDD or SSD_Index

Type

int

Default value

0x0

label13_write_stream_id

Label13: Write Stream ID

Type

int

Default value

0x0

label14_read_ssd

Label14: Read SDD or SSD_Index

Type

int

Default value

0x0

label14_read_stream_id

Label14: Read Stream ID

Type

int

Default value

0x0

label14_write_ssd

Label14: Write SDD or SSD_Index

Type

int

Default value

0x0

label14_write_stream_id

Label14: Write Stream ID

Type

int

Default value

0x0

label15_read_ssd

Label15: Read SDD or SSD_Index

Type

int

Default value

0x0

label15_read_stream_id

Label15: Read Stream ID

Type

int

Default value

0x0

label15_write_ssd

Label15: Write SDD or SSD_Index

Type

int

Default value

0x0

label15_write_stream_id

Label15: Write Stream ID

Type

int

Default value

0x0

label16_read_ssd

Label16: Read SDD or SSD_Index

Type

int

Default value

0x0

label16_read_stream_id

Label16: Read Stream ID

Type

int

Default value

0x0

label16_write_ssd

Label16: Write SDD or SSD_Index

Type

int

Default value

0x0

label16_write_stream_id

Label16: Write Stream ID

Type

int

Default value

0x0

label17_read_ssd

Label17: Read SDD or SSD_Index

Type

int

Default value

0x0

label17_read_stream_id

Label17: Read Stream ID

Type

int

Default value

0x0

label17_write_ssd

Label17: Write SDD or SSD_Index

Type

int

Default value

0x0

label17_write_stream_id

Label17: Write Stream ID

Type

int

Default value

0x0

label18_read_ssd

Label18: Read SDD or SSD_Index

Type

int

Default value

0x0

label18_read_stream_id

Label18: Read Stream ID

Type

int

Default value

0x0

label18_write_ssd

Label18: Write SDD or SSD_Index

Type

int

Default value

0x0

label18_write_stream_id

Label18: Write Stream ID

Type

int

Default value

0x0

label19_read_ssd

Label19: Read SDD or SSD_Index

Type

int

Default value

0x0

label19_read_stream_id

Label19: Read Stream ID

Type

int

Default value

0x0

label19_write_ssd

Label19: Write SDD or SSD_Index

Type

int

Default value

0x0

label19_write_stream_id

Label19: Write Stream ID

Type

int

Default value

0x0

label1_read_ssd

Label1: Read SDD or SSD_Index

Type

int

Default value

0x0

label1_read_stream_id

Label1: Read Stream ID

Type

int

Default value

0x0

label1_write_ssd

Label1: Write SDD or SSD_Index

Type

int

Default value

0x0

label1_write_stream_id

Label1: Write Stream ID

Type

int

Default value

0x0

label20_read_ssd

Label20: Read SDD or SSD_Index

Type

int

Default value

0x0

label20_read_stream_id

Label20: Read Stream ID

Type

int

Default value

0x0

label20_write_ssd

Label20: Write SDD or SSD_Index

Type

int

Default value

0x0

label20_write_stream_id

Label20: Write Stream ID

Type

int

Default value

0x0

label21_read_ssd

Label21: Read SDD or SSD_Index

Type

int

Default value

0x0

label21_read_stream_id

Label21: Read Stream ID

Type

int

Default value

0x0

label21_write_ssd

Label21: Write SDD or SSD_Index

Type

int

Default value

0x0

label21_write_stream_id

Label21: Write Stream ID

Type

int

Default value

0x0

label22_read_ssd

Label22: Read SDD or SSD_Index

Type

int

Default value

0x0

label22_read_stream_id

Label22: Read Stream ID

Type

int

Default value

0x0

label22_write_ssd

Label22: Write SDD or SSD_Index

Type

int

Default value

0x0

label22_write_stream_id

Label22: Write Stream ID

Type

int

Default value

0x0

label23_read_ssd

Label23: Read SDD or SSD_Index

Type

int

Default value

0x0

label23_read_stream_id

Label23: Read Stream ID

Type

int

Default value

0x0

label23_write_ssd

Label23: Write SDD or SSD_Index

Type

int

Default value

0x0

label23_write_stream_id

Label23: Write Stream ID

Type

int

Default value

0x0

label24_read_ssd

Label24: Read SDD or SSD_Index

Type

int

Default value

0x0

label24_read_stream_id

Label24: Read Stream ID

Type

int

Default value

0x0

label24_write_ssd

Label24: Write SDD or SSD_Index

Type

int

Default value

0x0

label24_write_stream_id

Label24: Write Stream ID

Type

int

Default value

0x0

label25_read_ssd

Label25: Read SDD or SSD_Index

Type

int

Default value

0x0

label25_read_stream_id

Label25: Read Stream ID

Type

int

Default value

0x0

label25_write_ssd

Label25: Write SDD or SSD_Index

Type

int

Default value

0x0

label25_write_stream_id

Label25: Write Stream ID

Type

int

Default value

0x0

label26_read_ssd

Label26: Read SDD or SSD_Index

Type

int

Default value

0x0

label26_read_stream_id

Label26: Read Stream ID

Type

int

Default value

0x0

label26_write_ssd

Label26: Write SDD or SSD_Index

Type

int

Default value

0x0

label26_write_stream_id

Label26: Write Stream ID

Type

int

Default value

0x0

label27_read_ssd

Label27: Read SDD or SSD_Index

Type

int

Default value

0x0

label27_read_stream_id

Label27: Read Stream ID

Type

int

Default value

0x0

label27_write_ssd

Label27: Write SDD or SSD_Index

Type

int

Default value

0x0

label27_write_stream_id

Label27: Write Stream ID

Type

int

Default value

0x0

label28_read_ssd

Label28: Read SDD or SSD_Index

Type

int

Default value

0x0

label28_read_stream_id

Label28: Read Stream ID

Type

int

Default value

0x0

label28_write_ssd

Label28: Write SDD or SSD_Index

Type

int

Default value

0x0

label28_write_stream_id

Label28: Write Stream ID

Type

int

Default value

0x0

label29_read_ssd

Label29: Read SDD or SSD_Index

Type

int

Default value

0x0

label29_read_stream_id

Label29: Read Stream ID

Type

int

Default value

0x0

label29_write_ssd

Label29: Write SDD or SSD_Index

Type

int

Default value

0x0

label29_write_stream_id

Label29: Write Stream ID

Type

int

Default value

0x0

label12_read_ssd

Label2: Read SDD or SSD_Index

Type

int

Default value

0x0

label12_read_stream_id

Label2: Read Stream ID

Type

int

Default value

0x0

label12_write_ssd

Label2: Write SDD or SSD_Index

Type

int

Default value

0x0

label2_write_stream_id

Label2: Write Stream ID

Type

int

Default value

0x0

label30_read_ssd

Label30: Read SDD or SSD_Index

Type

int

Default value

0x0

label30_read_stream_id

Label30: Read Stream ID

Type

int

Default value

0x0

label30_write_ssd

Label30: Write SDD or SSD_Index

Type

int

Default value

0x0

label30_write_stream_id

Label30: Write Stream ID

Type

int

Default value

0x0

label31_read_ssd

Label31: Read SDD or SSD_Index

Type

int

Default value

0x0

label31_read_stream_id

Label31: Read Stream ID

Type

int

Default value

0x0

label31_write_ssd

Label31: Write SDD or SSD_Index

Type

int

Default value

0x0

label31_write_stream_id

Label31: Write Stream ID

Type

int

Default value

0x0

label13_read_ssd

Label3: Read SDD or SSD_Index

Type

int

Default value

0x0

label13_read_stream_id

Label3: Read Stream ID

Type

int

Default value

0x0

label13_write_ssd

Label3: Write SDD or SSD_Index

Type

int

Default value

0x0

label3_write_stream_id

Label3: Write Stream ID

Type

int

Default value

0x0

label4_read_ssd

Label4: Read SDD or SSD_Index

Type

int

Default value

0x0

label4_read_stream_id

Label4: Read Stream ID

Type

int

Default value

0x0

label4_write_ssd

Label4: Write SDD or SSD_Index

Type

int

Default value

0x0

label4_write_stream_id

Label4: Write Stream ID

Type

int

Default value

0x0

label5_read_ssd

Label5: Read SDD or SSD_Index

Type

int

Default value

0x0

label5_read_stream_id

Label5: Read Stream ID

Type

int

Default value

0x0

label5_write_ssd

Label5: Write SDD or SSD_Index

Type

int

Default value

0x0

label5_write_stream_id

Label5: Write Stream ID

Type

int

Default value

0x0

label6_read_ssd

Label6: Read SDD or SSD_Index

Type

int

Default value

0x0

label6_read_stream_id

Label6: Read Stream ID

Type

int

Default value

0x0

label6_write_ssd

Label6: Write SDD or SSD_Index

Type

int

Default value

0x0

label6_write_stream_id

Label6: Write Stream ID

Type

int

Default value

0x0

label7_read_ssd

Label7: Read SDD or SSD_Index

Type

int

Default value

0x0

label7_read_stream_id

Label7: Read Stream ID

Type

int

Default value

0x0

label7_write_ssd

Label7: Write SDD or SSD_Index

Type

int

Default value

0x0

label7_write_stream_id

Label7: Write Stream ID

Type

int

Default value

0x0

label8_read_ssd

Label8: Read SDD or SSD_Index

Type

int

Default value

0x0

label8_read_stream_id

Label8: Read Stream ID

Type

int

Default value

0x0

label8_write_ssd

Label8: Write SDD or SSD_Index

Type

int

Default value

0x0

label8_write_stream_id

Label8: Write Stream ID

Type

int

Default value

0x0

label9_read_ssd

Label9: Read SDD or SSD_Index

Type

int

Default value

0x0

label9_read_stream_id

Label9: Read Stream ID

Type

int

Default value

0x0

label9_write_ssd

Label9: Write SDD or SSD_Index

Type

int

Default value

0x0

label9_write_stream_id

Label9: Write Stream ID

Type

int

Default value

0x0

mmu.PRIVATE_PARAMETER_personality

The personality to use (affects ID codes and various imp def features).

Type

string

Default value**mmu.PRIVATE_PARAMETER_seed**

Seed for randomised SMMU implementation defined behaviour

Type

int

Default value

0x12345678

mmu.PRIVATE_PARAMETER_validation_mode

Internal validation mode.

Type

int

Default value

0x0

number_of_contexts

Number of context banks

Type

int

Default value

0x8

number_of_smrs

Number of stream match registers.

Type

int

Default value

0x20

percent_tlbstatus_commits

Percentage of times that a poll of TLBSTATUS will commit the TLBI commands

Type

int

Default value

0xa

prefetch_only_requests

Handle prefetch-only requests by:- 0 -- deny them 1 -- use debug table walks and TLB entries 2 -- treat them as normal transactions (dangerous)

Type

int

Default value

0x0

programmable_non_secure_by_default_ssd_indices

Programmable SSD Indexes that are by default non-secure (e.g. 0, 6, 35-84).

Type

string

Default value**programmable_secure_by_default_ssd_indices**

Programmable SSD Indexes that are by default secure (e.g. 0, 6, 35-84).

Type

string

Default value**ptw_has_separate_port**

Page Table Walks use pvbus_ptw_m

Type

bool

Default value

0x1

supports_nested_translations

Supports nested translations (stage 1 + stage 2)

Type

bool

Default value

0x1

- tlb_depth**

TLB Depth (0 means 10000). The model will perform best with more TLB entries.

Type

int

Default value

0x800
- use_label_mapping**

Use label mapping

Type

bool

Default value

0x1
- use_ssd_determination_table**

Use SSD Determination Table

Type

bool

Default value

0x1
- version**

Version of the RTL that the model represents. Valid values are LACr1 and EAC.

Type

string

Default value

EAC

3.10.53 MMU_500_BASE

MMU-500 base component. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1042: IP revisions support

Revision	Quality level
r2p4	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for MMU_500_BASE

This model has the following Iris instances:

Table 3-1043: MMU_500_BASE Iris instances

InstanceName	ComponentName
MMU_500_BASE	MMU_500_BASE
MMU_500_BASE.mapper0	PVBusMapper
MMU_500_BASE.mapper1	PVBusMapper
MMU_500_BASE.mapper10	PVBusMapper
MMU_500_BASE.mapper11	PVBusMapper
MMU_500_BASE.mapper12	PVBusMapper
MMU_500_BASE.mapper13	PVBusMapper
MMU_500_BASE.mapper14	PVBusMapper
MMU_500_BASE.mapper15	PVBusMapper
MMU_500_BASE.mapper16	PVBusMapper
MMU_500_BASE.mapper17	PVBusMapper
MMU_500_BASE.mapper18	PVBusMapper
MMU_500_BASE.mapper19	PVBusMapper
MMU_500_BASE.mapper2	PVBusMapper
MMU_500_BASE.mapper20	PVBusMapper
MMU_500_BASE.mapper21	PVBusMapper
MMU_500_BASE.mapper22	PVBusMapper
MMU_500_BASE.mapper23	PVBusMapper
MMU_500_BASE.mapper24	PVBusMapper
MMU_500_BASE.mapper25	PVBusMapper
MMU_500_BASE.mapper26	PVBusMapper
MMU_500_BASE.mapper27	PVBusMapper
MMU_500_BASE.mapper28	PVBusMapper
MMU_500_BASE.mapper29	PVBusMapper
MMU_500_BASE.mapper3	PVBusMapper
MMU_500_BASE.mapper30	PVBusMapper
MMU_500_BASE.mapper31	PVBusMapper
MMU_500_BASE.mapper4	PVBusMapper
MMU_500_BASE.mapper5	PVBusMapper
MMU_500_BASE.mapper6	PVBusMapper
MMU_500_BASE.mapper7	PVBusMapper
MMU_500_BASE.mapper8	PVBusMapper
MMU_500_BASE.mapper9	PVBusMapper
MMU_500_BASE.ptw_dvm_receiver	PVBusMapper
MMU_500_BASE.ptw_master	PVBusMaster
MMU_500_BASE.pvbus_control_s_slv	PVBusSlave
MMU_500_BASE.pvbus_master	PVBusMaster
MMU_500_BASE.pvbusmaster0	PVBusMaster

InstanceName	ComponentName
MMU_500_BASE.pvbusmaster1	PVBusMaster
MMU_500_BASE.pvbusmaster10	PVBusMaster
MMU_500_BASE.pvbusmaster11	PVBusMaster
MMU_500_BASE.pvbusmaster12	PVBusMaster
MMU_500_BASE.pvbusmaster13	PVBusMaster
MMU_500_BASE.pvbusmaster14	PVBusMaster
MMU_500_BASE.pvbusmaster15	PVBusMaster
MMU_500_BASE.pvbusmaster16	PVBusMaster
MMU_500_BASE.pvbusmaster17	PVBusMaster
MMU_500_BASE.pvbusmaster18	PVBusMaster
MMU_500_BASE.pvbusmaster19	PVBusMaster
MMU_500_BASE.pvbusmaster2	PVBusMaster
MMU_500_BASE.pvbusmaster20	PVBusMaster
MMU_500_BASE.pvbusmaster21	PVBusMaster
MMU_500_BASE.pvbusmaster22	PVBusMaster
MMU_500_BASE.pvbusmaster23	PVBusMaster
MMU_500_BASE.pvbusmaster24	PVBusMaster
MMU_500_BASE.pvbusmaster25	PVBusMaster
MMU_500_BASE.pvbusmaster26	PVBusMaster
MMU_500_BASE.pvbusmaster27	PVBusMaster
MMU_500_BASE.pvbusmaster28	PVBusMaster
MMU_500_BASE.pvbusmaster29	PVBusMaster
MMU_500_BASE.pvbusmaster3	PVBusMaster
MMU_500_BASE.pvbusmaster30	PVBusMaster
MMU_500_BASE.pvbusmaster31	PVBusMaster
MMU_500_BASE.pvbusmaster4	PVBusMaster
MMU_500_BASE.pvbusmaster5	PVBusMaster
MMU_500_BASE.pvbusmaster6	PVBusMaster
MMU_500_BASE.pvbusmaster7	PVBusMaster
MMU_500_BASE.pvbusmaster8	PVBusMaster
MMU_500_BASE.pvbusmaster9	PVBusMaster
MMU_500_BASE.pvbusslave0	PVBusSlave
MMU_500_BASE.pvbusslave1	PVBusSlave
MMU_500_BASE.pvbusslave10	PVBusSlave
MMU_500_BASE.pvbusslave11	PVBusSlave
MMU_500_BASE.pvbusslave12	PVBusSlave
MMU_500_BASE.pvbusslave13	PVBusSlave
MMU_500_BASE.pvbusslave14	PVBusSlave
MMU_500_BASE.pvbusslave15	PVBusSlave

InstanceName	ComponentName
MMU_500_BASE.pvbuslave16	PVBusSlave
MMU_500_BASE.pvbuslave17	PVBusSlave
MMU_500_BASE.pvbuslave18	PVBusSlave
MMU_500_BASE.pvbuslave19	PVBusSlave
MMU_500_BASE.pvbuslave2	PVBusSlave
MMU_500_BASE.pvbuslave20	PVBusSlave
MMU_500_BASE.pvbuslave21	PVBusSlave
MMU_500_BASE.pvbuslave22	PVBusSlave
MMU_500_BASE.pvbuslave23	PVBusSlave
MMU_500_BASE.pvbuslave24	PVBusSlave
MMU_500_BASE.pvbuslave25	PVBusSlave
MMU_500_BASE.pvbuslave26	PVBusSlave
MMU_500_BASE.pvbuslave27	PVBusSlave
MMU_500_BASE.pvbuslave28	PVBusSlave
MMU_500_BASE.pvbuslave29	PVBusSlave
MMU_500_BASE.pvbuslave3	PVBusSlave
MMU_500_BASE.pvbuslave30	PVBusSlave
MMU_500_BASE.pvbuslave31	PVBusSlave
MMU_500_BASE.pvbuslave4	PVBusSlave
MMU_500_BASE.pvbuslave5	PVBusSlave
MMU_500_BASE.pvbuslave6	PVBusSlave
MMU_500_BASE.pvbuslave7	PVBusSlave
MMU_500_BASE.pvbuslave8	PVBusSlave
MMU_500_BASE.pvbuslave9	PVBusSlave

This model has the following MTI trace components:

Table 3-1044: MMU_500_BASE MTI instances

InstanceName	ComponentName
MMU_500_BASE	MMU_500_BASE
MMU_500_BASE.mapper0	PVBusMapper
MMU_500_BASE.mapper1	PVBusMapper
MMU_500_BASE.mapper10	PVBusMapper
MMU_500_BASE.mapper11	PVBusMapper
MMU_500_BASE.mapper12	PVBusMapper
MMU_500_BASE.mapper13	PVBusMapper
MMU_500_BASE.mapper14	PVBusMapper
MMU_500_BASE.mapper15	PVBusMapper
MMU_500_BASE.mapper16	PVBusMapper
MMU_500_BASE.mapper17	PVBusMapper

InstanceName	ComponentName
MMU_500_BASE.mapper18	PVBusMapper
MMU_500_BASE.mapper19	PVBusMapper
MMU_500_BASE.mapper2	PVBusMapper
MMU_500_BASE.mapper20	PVBusMapper
MMU_500_BASE.mapper21	PVBusMapper
MMU_500_BASE.mapper22	PVBusMapper
MMU_500_BASE.mapper23	PVBusMapper
MMU_500_BASE.mapper24	PVBusMapper
MMU_500_BASE.mapper25	PVBusMapper
MMU_500_BASE.mapper26	PVBusMapper
MMU_500_BASE.mapper27	PVBusMapper
MMU_500_BASE.mapper28	PVBusMapper
MMU_500_BASE.mapper29	PVBusMapper
MMU_500_BASE.mapper3	PVBusMapper
MMU_500_BASE.mapper30	PVBusMapper
MMU_500_BASE.mapper31	PVBusMapper
MMU_500_BASE.mapper4	PVBusMapper
MMU_500_BASE.mapper5	PVBusMapper
MMU_500_BASE.mapper6	PVBusMapper
MMU_500_BASE.mapper7	PVBusMapper
MMU_500_BASE.mapper8	PVBusMapper
MMU_500_BASE.mapper9	PVBusMapper
MMU_500_BASE.ptw_dvm_receiver	PVBusMapper
MMU_500_BASE.ptw_master	PVBusMaster
MMU_500_BASE.pvbus_control_s_slv	PVBusSlave
MMU_500_BASE.pvbus_master	PVBusMaster
MMU_500_BASE.pvbusmaster0	PVBusMaster
MMU_500_BASE.pvbusmaster1	PVBusMaster
MMU_500_BASE.pvbusmaster10	PVBusMaster
MMU_500_BASE.pvbusmaster11	PVBusMaster
MMU_500_BASE.pvbusmaster12	PVBusMaster
MMU_500_BASE.pvbusmaster13	PVBusMaster
MMU_500_BASE.pvbusmaster14	PVBusMaster
MMU_500_BASE.pvbusmaster15	PVBusMaster
MMU_500_BASE.pvbusmaster16	PVBusMaster
MMU_500_BASE.pvbusmaster17	PVBusMaster
MMU_500_BASE.pvbusmaster18	PVBusMaster
MMU_500_BASE.pvbusmaster19	PVBusMaster
MMU_500_BASE.pvbusmaster2	PVBusMaster

InstanceName	ComponentName
MMU_500_BASE.pvbusmaster20	PVBusMaster
MMU_500_BASE.pvbusmaster21	PVBusMaster
MMU_500_BASE.pvbusmaster22	PVBusMaster
MMU_500_BASE.pvbusmaster23	PVBusMaster
MMU_500_BASE.pvbusmaster24	PVBusMaster
MMU_500_BASE.pvbusmaster25	PVBusMaster
MMU_500_BASE.pvbusmaster26	PVBusMaster
MMU_500_BASE.pvbusmaster27	PVBusMaster
MMU_500_BASE.pvbusmaster28	PVBusMaster
MMU_500_BASE.pvbusmaster29	PVBusMaster
MMU_500_BASE.pvbusmaster3	PVBusMaster
MMU_500_BASE.pvbusmaster30	PVBusMaster
MMU_500_BASE.pvbusmaster31	PVBusMaster
MMU_500_BASE.pvbusmaster4	PVBusMaster
MMU_500_BASE.pvbusmaster5	PVBusMaster
MMU_500_BASE.pvbusmaster6	PVBusMaster
MMU_500_BASE.pvbusmaster7	PVBusMaster
MMU_500_BASE.pvbusmaster8	PVBusMaster
MMU_500_BASE.pvbusmaster9	PVBusMaster
MMU_500_BASE.pvbusslave0	PVBusSlave
MMU_500_BASE.pvbusslave1	PVBusSlave
MMU_500_BASE.pvbusslave10	PVBusSlave
MMU_500_BASE.pvbusslave11	PVBusSlave
MMU_500_BASE.pvbusslave12	PVBusSlave
MMU_500_BASE.pvbusslave13	PVBusSlave
MMU_500_BASE.pvbusslave14	PVBusSlave
MMU_500_BASE.pvbusslave15	PVBusSlave
MMU_500_BASE.pvbusslave16	PVBusSlave
MMU_500_BASE.pvbusslave17	PVBusSlave
MMU_500_BASE.pvbusslave18	PVBusSlave
MMU_500_BASE.pvbusslave19	PVBusSlave
MMU_500_BASE.pvbusslave2	PVBusSlave
MMU_500_BASE.pvbusslave20	PVBusSlave
MMU_500_BASE.pvbusslave21	PVBusSlave
MMU_500_BASE.pvbusslave22	PVBusSlave
MMU_500_BASE.pvbusslave23	PVBusSlave
MMU_500_BASE.pvbusslave24	PVBusSlave
MMU_500_BASE.pvbusslave25	PVBusSlave
MMU_500_BASE.pvbusslave26	PVBusSlave

InstanceName	ComponentName
MMU_500_BASE.pvbuslave27	PVBusSlave
MMU_500_BASE.pvbuslave28	PVBusSlave
MMU_500_BASE.pvbuslave29	PVBusSlave
MMU_500_BASE.pvbuslave3	PVBusSlave
MMU_500_BASE.pvbuslave30	PVBusSlave
MMU_500_BASE.pvbuslave31	PVBusSlave
MMU_500_BASE.pvbuslave4	PVBusSlave
MMU_500_BASE.pvbuslave5	PVBusSlave
MMU_500_BASE.pvbuslave6	PVBusSlave
MMU_500_BASE.pvbuslave7	PVBusSlave
MMU_500_BASE.pvbuslave8	PVBusSlave
MMU_500_BASE.pvbuslave9	PVBusSlave

MMU_500_BASE contains the following CADI targets:

- MMU_500_BASE

Ports for MMU_500_BASE

Table 3-1045: Ports

Name	Protocol	Type	Description
cfg_cttw_in	Signal	Slave	The SoC supports coherent page walks, this is meant to be sampled at reset. However, in practice the model has to prevent the race condition between cfg_cttw being asserted at the same 'cycle' as negedge reset. Thus we actually only sample the signal on the first transaction to the SMMU or the first transition on this signal after reset. Thus in the model, we require that cfg_cttw be held for at least this period of time.
comb_irpt_ns	Signal	Master	"Non-secure combined interrupt"
comb_irpt_s	Signal	Master	"Secure combined interrupt"
cxt_irpt[128]	Signal	Master	Non-secure context bank fault.
glblflt_irpt_ns	Signal	Master	Global non-secure Fault Interrupt In the SMMU Architecture this is called SMMU_NSglrpt.
glblflt_irpt_s	Signal	Master	Global secure Fault Interrupt In the SMMU Architecture this is called SMMU_glrlpt.
identify	MMU_500_BASE_IDENTIFY	Master	This port is a special model port that is used to take a transaction and map it to an SSD/SSD_Index and StreamID.
priv_internals	MMU_500_Internals	Slave	For internal use only, please do not use.
pvbus_control_s	PVBus	Slave	The register port of the device is AXI.
pvbus_m[32]	PVBus	Master	This downstream port is where the translated accesses from pvbus_s emerge. See notes for pvbus_s[] as well. If the Page Table Walk (PTW) does not have a separate port then PTW accesses will emerge at port 0 with the same attributes as described in pvbus_ptw_m.

Name	Protocol	Type	Description
pvbus_ptw_m	PVBus	Master	This downstream port is where page table walk accesses come from. This is only used if configured to use a separate page table walk port. The MMU-500 will only obey DVM messages if configured to use this port. The page walks come out of this port with the following master_id and user_flags. master_id : 0xFFFFFFFF The user flags : user_flags[7:0] stage 1 context_id (or 0xFF if stage2 only) user_flags[15:8] stage 2 context_id (or 0xFF if stage 1 with stage 2 bypass) user_flags[18:16] stage 1 level user_flags[21:19] stage 2 level user_flags[31,30] adomain of the transaction NOTE that if the walk is being done for a stage 1 page walk descriptor fetch then the stage 1 level field will indicate that level. If the walk is being done for a stage 2 descriptor fetch, then the stage 2 level field will show that level. If the context-id for a stage is not valid (0xFF) then the 'level' information is 0x7.
pvbus_s[32]	PVBus	Slave	This port is the upstream port of the device, addresses on the port are in the VA/IPA Each TBU in the design is represented by a pair of pvbus_s[tbu_id] and pvbus_m[tbu_id]. That is transactions that go into pvbus_s[tbu_id] will emerge at pvbus_m[tbu_id]. The port index that a transaction comes in on is the tbu_number_ parameter to the MMU_500_BASE_IDENTIFY::identify() function. The identify() function must use all the information it is given by the parameters to map to the architectural concepts of StreamID and SSD_Index/SSD. How it does this is IMPLEMENTATION DEFINED and depends on the topology of the SoC and the masters upstream of the TBUs.
reset_in	Signal	Slave	The reset pin.

Parameters for MMU_500_BASE

PRIVATE_PARAMETER_personality

The personality to use (affects ID codes and various imp def features).

Type

string

Default value

PRIVATE_PARAMETER_seed

Seed for randomised SMMU implementation defined behaviour

Type

int

Default value

0x12345678

PRIVATE_PARAMETER_validation_mode

Internal validation mode.

Type

int

Default value

0x0

always_secure_ssd_indices

Non-programmable SSD Indexes that are always secure (e.g. 0, 6, 35-64).

Type

string

Default value**cfg_cttw**

Perform coherent page table walks

Type

bool

Default value

0x1

dump_unpredictability_in_user_flags

Override the user flags to encode unpredictable information (validation only)

Type

bool

Default value

0x0

number_of_contexts

Number of context banks

Type

int

Default value

0x8

number_of_smrs

Number of stream match registers.

Type

int

Default value

0x10

percent_tlbstatus_commits

Percentage of times that a poll of TLBSTATUS will commit the TLBI commands

Type

int

Default value

0xa

prefetch_only_requests

Handle prefetch-only requests by:- 0 -- deny them 1 -- use debug table walks and TLB entries 2 -- treat them as normal transactions (dangerous)

Type

int

Default value

0x0

programmable_non_secure_by_default_ssd_indices

Programmable SSD Indexes that are by default non-secure (e.g. 0, 6, 35-84).

Type

string

Default value**programmable_secure_by_default_ssd_indices**

Programmable SSD Indexes that are by default secure (e.g. 0, 6, 35-84).

Type

string

Default value**ptw_has_separate_port**

Page Table Walks use pvbus_ptw_m (or uses pvbus_m[0])

Type

bool

Default value

0x1

supports_nested_translations

Supports nested translations (stage 1 + stage 2)

Type

bool

Default value

0x1

tlb_depth

TLB Depth (0 means 10000). The model will perform best with more TLB entries.

Type

int

Default value

0x800

use_ssd_determination_table
Use SSD Determination Table

Type
bool

Default value
0x1

version
Version of the RTL that the model represents. Valid values are LACr1 and EAC.

Type
string

Default value
EAC

3.10.54 MMU_600

This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1046: IP revisions support

Revision	Quality level
r0p0	Full support
r0p1	Full support
r0p2	Full support
r1p0	Full support
r2p0	Full support
r2p1	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- sup_btm
- sup_sev

Iris and MTI instances for MMU_600

This model has the following Iris instances:

Table 3-1047: MMU_600 Iris instances

InstanceName	ComponentName
MMU_600	MMU_600
MMU_600.register_file[0]	PVBusSlave
MMU_600.service_request_tbu[0]	PVBusSlave
MMU_600.service_request_tbu[10]	PVBusSlave
MMU_600.service_request_tbu[11]	PVBusSlave
MMU_600.service_request_tbu[12]	PVBusSlave
MMU_600.service_request_tbu[13]	PVBusSlave
MMU_600.service_request_tbu[14]	PVBusSlave
MMU_600.service_request_tbu[15]	PVBusSlave
MMU_600.service_request_tbu[16]	PVBusSlave
MMU_600.service_request_tbu[17]	PVBusSlave
MMU_600.service_request_tbu[18]	PVBusSlave
MMU_600.service_request_tbu[19]	PVBusSlave
MMU_600.service_request_tbu[1]	PVBusSlave
MMU_600.service_request_tbu[20]	PVBusSlave
MMU_600.service_request_tbu[21]	PVBusSlave
MMU_600.service_request_tbu[22]	PVBusSlave
MMU_600.service_request_tbu[23]	PVBusSlave
MMU_600.service_request_tbu[24]	PVBusSlave
MMU_600.service_request_tbu[25]	PVBusSlave
MMU_600.service_request_tbu[26]	PVBusSlave
MMU_600.service_request_tbu[27]	PVBusSlave
MMU_600.service_request_tbu[28]	PVBusSlave
MMU_600.service_request_tbu[29]	PVBusSlave
MMU_600.service_request_tbu[2]	PVBusSlave
MMU_600.service_request_tbu[30]	PVBusSlave
MMU_600.service_request_tbu[31]	PVBusSlave
MMU_600.service_request_tbu[32]	PVBusSlave
MMU_600.service_request_tbu[33]	PVBusSlave
MMU_600.service_request_tbu[34]	PVBusSlave
MMU_600.service_request_tbu[35]	PVBusSlave
MMU_600.service_request_tbu[36]	PVBusSlave
MMU_600.service_request_tbu[37]	PVBusSlave
MMU_600.service_request_tbu[38]	PVBusSlave
MMU_600.service_request_tbu[39]	PVBusSlave
MMU_600.service_request_tbu[3]	PVBusSlave
MMU_600.service_request_tbu[40]	PVBusSlave
MMU_600.service_request_tbu[41]	PVBusSlave

InstanceName	ComponentName
MMU_600.service_request_tbu[42]	PVBusSlave
MMU_600.service_request_tbu[43]	PVBusSlave
MMU_600.service_request_tbu[44]	PVBusSlave
MMU_600.service_request_tbu[45]	PVBusSlave
MMU_600.service_request_tbu[46]	PVBusSlave
MMU_600.service_request_tbu[47]	PVBusSlave
MMU_600.service_request_tbu[48]	PVBusSlave
MMU_600.service_request_tbu[49]	PVBusSlave
MMU_600.service_request_tbu[4]	PVBusSlave
MMU_600.service_request_tbu[50]	PVBusSlave
MMU_600.service_request_tbu[51]	PVBusSlave
MMU_600.service_request_tbu[52]	PVBusSlave
MMU_600.service_request_tbu[53]	PVBusSlave
MMU_600.service_request_tbu[54]	PVBusSlave
MMU_600.service_request_tbu[55]	PVBusSlave
MMU_600.service_request_tbu[56]	PVBusSlave
MMU_600.service_request_tbu[57]	PVBusSlave
MMU_600.service_request_tbu[58]	PVBusSlave
MMU_600.service_request_tbu[59]	PVBusSlave
MMU_600.service_request_tbu[5]	PVBusSlave
MMU_600.service_request_tbu[60]	PVBusSlave
MMU_600.service_request_tbu[61]	PVBusSlave
MMU_600.service_request_tbu[62]	PVBusSlave
MMU_600.service_request_tbu[63]	PVBusSlave
MMU_600.service_request_tbu[6]	PVBusSlave
MMU_600.service_request_tbu[7]	PVBusSlave
MMU_600.service_request_tbu[8]	PVBusSlave
MMU_600.service_request_tbu[9]	PVBusSlave
MMU_600.tbu[0]	PVBusMapper
MMU_600.tbu[10]	PVBusMapper
MMU_600.tbu[11]	PVBusMapper
MMU_600.tbu[12]	PVBusMapper
MMU_600.tbu[13]	PVBusMapper
MMU_600.tbu[14]	PVBusMapper
MMU_600.tbu[15]	PVBusMapper
MMU_600.tbu[16]	PVBusMapper
MMU_600.tbu[17]	PVBusMapper
MMU_600.tbu[18]	PVBusMapper
MMU_600.tbu[19]	PVBusMapper

InstanceName	ComponentName
MMU_600.tbu[1]	PVBusMapper
MMU_600.tbu[20]	PVBusMapper
MMU_600.tbu[21]	PVBusMapper
MMU_600.tbu[22]	PVBusMapper
MMU_600.tbu[23]	PVBusMapper
MMU_600.tbu[24]	PVBusMapper
MMU_600.tbu[25]	PVBusMapper
MMU_600.tbu[26]	PVBusMapper
MMU_600.tbu[27]	PVBusMapper
MMU_600.tbu[28]	PVBusMapper
MMU_600.tbu[29]	PVBusMapper
MMU_600.tbu[2]	PVBusMapper
MMU_600.tbu[30]	PVBusMapper
MMU_600.tbu[31]	PVBusMapper
MMU_600.tbu[32]	PVBusMapper
MMU_600.tbu[33]	PVBusMapper
MMU_600.tbu[34]	PVBusMapper
MMU_600.tbu[35]	PVBusMapper
MMU_600.tbu[36]	PVBusMapper
MMU_600.tbu[37]	PVBusMapper
MMU_600.tbu[38]	PVBusMapper
MMU_600.tbu[39]	PVBusMapper
MMU_600.tbu[3]	PVBusMapper
MMU_600.tbu[40]	PVBusMapper
MMU_600.tbu[41]	PVBusMapper
MMU_600.tbu[42]	PVBusMapper
MMU_600.tbu[43]	PVBusMapper
MMU_600.tbu[44]	PVBusMapper
MMU_600.tbu[45]	PVBusMapper
MMU_600.tbu[46]	PVBusMapper
MMU_600.tbu[47]	PVBusMapper
MMU_600.tbu[48]	PVBusMapper
MMU_600.tbu[49]	PVBusMapper
MMU_600.tbu[4]	PVBusMapper
MMU_600.tbu[50]	PVBusMapper
MMU_600.tbu[51]	PVBusMapper
MMU_600.tbu[52]	PVBusMapper
MMU_600.tbu[53]	PVBusMapper
MMU_600.tbu[54]	PVBusMapper

InstanceName	ComponentName
MMU_600.tb[55]	PVBusMapper
MMU_600.tb[56]	PVBusMapper
MMU_600.tb[57]	PVBusMapper
MMU_600.tb[58]	PVBusMapper
MMU_600.tb[59]	PVBusMapper
MMU_600.tb[5]	PVBusMapper
MMU_600.tb[60]	PVBusMapper
MMU_600.tb[61]	PVBusMapper
MMU_600.tb[62]	PVBusMapper
MMU_600.tb[63]	PVBusMapper
MMU_600.tb[6]	PVBusMapper
MMU_600.tb[7]	PVBusMapper
MMU_600.tb[8]	PVBusMapper
MMU_600.tb[9]	PVBusMapper

This model has the following MTI trace components:

Table 3-1048: MMU_600 MTI instances

InstanceName	ComponentName
MMU_600	MMU_600
MMU_600.register_file[0]	PVBusSlave
MMU_600.service_request_tbu[0]	PVBusSlave
MMU_600.service_request_tbu[10]	PVBusSlave
MMU_600.service_request_tbu[11]	PVBusSlave
MMU_600.service_request_tbu[12]	PVBusSlave
MMU_600.service_request_tbu[13]	PVBusSlave
MMU_600.service_request_tbu[14]	PVBusSlave
MMU_600.service_request_tbu[15]	PVBusSlave
MMU_600.service_request_tbu[16]	PVBusSlave
MMU_600.service_request_tbu[17]	PVBusSlave
MMU_600.service_request_tbu[18]	PVBusSlave
MMU_600.service_request_tbu[19]	PVBusSlave
MMU_600.service_request_tbu[1]	PVBusSlave
MMU_600.service_request_tbu[20]	PVBusSlave
MMU_600.service_request_tbu[21]	PVBusSlave
MMU_600.service_request_tbu[22]	PVBusSlave
MMU_600.service_request_tbu[23]	PVBusSlave
MMU_600.service_request_tbu[24]	PVBusSlave
MMU_600.service_request_tbu[25]	PVBusSlave
MMU_600.service_request_tbu[26]	PVBusSlave

InstanceName	ComponentName
MMU_600.service_request_tbu[27]	PVBusSlave
MMU_600.service_request_tbu[28]	PVBusSlave
MMU_600.service_request_tbu[29]	PVBusSlave
MMU_600.service_request_tbu[2]	PVBusSlave
MMU_600.service_request_tbu[30]	PVBusSlave
MMU_600.service_request_tbu[31]	PVBusSlave
MMU_600.service_request_tbu[32]	PVBusSlave
MMU_600.service_request_tbu[33]	PVBusSlave
MMU_600.service_request_tbu[34]	PVBusSlave
MMU_600.service_request_tbu[35]	PVBusSlave
MMU_600.service_request_tbu[36]	PVBusSlave
MMU_600.service_request_tbu[37]	PVBusSlave
MMU_600.service_request_tbu[38]	PVBusSlave
MMU_600.service_request_tbu[39]	PVBusSlave
MMU_600.service_request_tbu[3]	PVBusSlave
MMU_600.service_request_tbu[40]	PVBusSlave
MMU_600.service_request_tbu[41]	PVBusSlave
MMU_600.service_request_tbu[42]	PVBusSlave
MMU_600.service_request_tbu[43]	PVBusSlave
MMU_600.service_request_tbu[44]	PVBusSlave
MMU_600.service_request_tbu[45]	PVBusSlave
MMU_600.service_request_tbu[46]	PVBusSlave
MMU_600.service_request_tbu[47]	PVBusSlave
MMU_600.service_request_tbu[48]	PVBusSlave
MMU_600.service_request_tbu[49]	PVBusSlave
MMU_600.service_request_tbu[4]	PVBusSlave
MMU_600.service_request_tbu[50]	PVBusSlave
MMU_600.service_request_tbu[51]	PVBusSlave
MMU_600.service_request_tbu[52]	PVBusSlave
MMU_600.service_request_tbu[53]	PVBusSlave
MMU_600.service_request_tbu[54]	PVBusSlave
MMU_600.service_request_tbu[55]	PVBusSlave
MMU_600.service_request_tbu[56]	PVBusSlave
MMU_600.service_request_tbu[57]	PVBusSlave
MMU_600.service_request_tbu[58]	PVBusSlave
MMU_600.service_request_tbu[59]	PVBusSlave
MMU_600.service_request_tbu[5]	PVBusSlave
MMU_600.service_request_tbu[60]	PVBusSlave
MMU_600.service_request_tbu[61]	PVBusSlave

InstanceName	ComponentName
MMU_600.service_request_tbu[62]	PVBusSlave
MMU_600.service_request_tbu[63]	PVBusSlave
MMU_600.service_request_tbu[6]	PVBusSlave
MMU_600.service_request_tbu[7]	PVBusSlave
MMU_600.service_request_tbu[8]	PVBusSlave
MMU_600.service_request_tbu[9]	PVBusSlave
MMU_600.tbu[0]	PVBusMapper
MMU_600.tbu[10]	PVBusMapper
MMU_600.tbu[11]	PVBusMapper
MMU_600.tbu[12]	PVBusMapper
MMU_600.tbu[13]	PVBusMapper
MMU_600.tbu[14]	PVBusMapper
MMU_600.tbu[15]	PVBusMapper
MMU_600.tbu[16]	PVBusMapper
MMU_600.tbu[17]	PVBusMapper
MMU_600.tbu[18]	PVBusMapper
MMU_600.tbu[19]	PVBusMapper
MMU_600.tbu[1]	PVBusMapper
MMU_600.tbu[20]	PVBusMapper
MMU_600.tbu[21]	PVBusMapper
MMU_600.tbu[22]	PVBusMapper
MMU_600.tbu[23]	PVBusMapper
MMU_600.tbu[24]	PVBusMapper
MMU_600.tbu[25]	PVBusMapper
MMU_600.tbu[26]	PVBusMapper
MMU_600.tbu[27]	PVBusMapper
MMU_600.tbu[28]	PVBusMapper
MMU_600.tbu[29]	PVBusMapper
MMU_600.tbu[2]	PVBusMapper
MMU_600.tbu[30]	PVBusMapper
MMU_600.tbu[31]	PVBusMapper
MMU_600.tbu[32]	PVBusMapper
MMU_600.tbu[33]	PVBusMapper
MMU_600.tbu[34]	PVBusMapper
MMU_600.tbu[35]	PVBusMapper
MMU_600.tbu[36]	PVBusMapper
MMU_600.tbu[37]	PVBusMapper
MMU_600.tbu[38]	PVBusMapper
MMU_600.tbu[39]	PVBusMapper

InstanceName	ComponentName
MMU_600.tbu[3]	PVBusMapper
MMU_600.tbu[40]	PVBusMapper
MMU_600.tbu[41]	PVBusMapper
MMU_600.tbu[42]	PVBusMapper
MMU_600.tbu[43]	PVBusMapper
MMU_600.tbu[44]	PVBusMapper
MMU_600.tbu[45]	PVBusMapper
MMU_600.tbu[46]	PVBusMapper
MMU_600.tbu[47]	PVBusMapper
MMU_600.tbu[48]	PVBusMapper
MMU_600.tbu[49]	PVBusMapper
MMU_600.tbu[4]	PVBusMapper
MMU_600.tbu[50]	PVBusMapper
MMU_600.tbu[51]	PVBusMapper
MMU_600.tbu[52]	PVBusMapper
MMU_600.tbu[53]	PVBusMapper
MMU_600.tbu[54]	PVBusMapper
MMU_600.tbu[55]	PVBusMapper
MMU_600.tbu[56]	PVBusMapper
MMU_600.tbu[57]	PVBusMapper
MMU_600.tbu[58]	PVBusMapper
MMU_600.tbu[59]	PVBusMapper
MMU_600.tbu[5]	PVBusMapper
MMU_600.tbu[60]	PVBusMapper
MMU_600.tbu[61]	PVBusMapper
MMU_600.tbu[62]	PVBusMapper
MMU_600.tbu[63]	PVBusMapper
MMU_600.tbu[6]	PVBusMapper
MMU_600.tbu[7]	PVBusMapper
MMU_600.tbu[8]	PVBusMapper
MMU_600.tbu[9]	PVBusMapper

MMU_600 contains the following CADI targets:

- MMU_600

About MMU_600

MMU_600 and MMU_700 are SMMUv3-compliant devices and are used for I/O virtualization of devices.

The hardware is a distributed SMMU. It consists of the following:

- A single Translation Control Unit (TCU), which has a port for the programming interface of the SMMU, receives DVM messages, and does all the page walking and queue manipulation, for example.
- One or more Translation Bus Units (TBUs), which translate transactions from upstream client devices into downstream transactions.
- Zero or more connections to PCIe Root Complexes (PCIe-RCs). A maximum of 62 TBUs and PCIe-RCs can be attached.
- An interconnect that connects the TBUs and PCIe-RCs to the TCU.

A PCIe-RC has one connection to the TCU to make ATS requests but the PCIe-RC uses one or more TBUs to transform the transactions and pass them to the memory system. In the model, the TBUs used are listed in the parameter `list_of_pcie_mode`. The SMMU does not know which TBUs a particular PCIe-RC is attached to.

The TBU ORs a value into each StreamID that it receives. In the model, this is configured by the following parameters:

- `list_of_s_sid_high_at_bitpos0`
- `list_of_ns_sid_high_at_bitpos0`

The TCU, TBU, and the interconnect are all represented by this single model component.

In the model, the `tbs_pvbus_s[i]` and `tbm_pvbus_m[i]` port pair represent a TBU `i`, or `tbs_pvbus_s[i]` represents an incoming connection for a PCIe-RC. The corresponding reverse connection from the TCU to the PCIe-RC is by a special bus called `pvbus_id_routed_m` that is used to transport ATC Invalidates to the PCIe-RC.

To reduce system construction complexity, the `tbs_pvbus_s[i]` and `tbm_pvbus_m[i]` pair also acts as a TBU so that the PCIe-RC does not need to separate its normal transactions and its ATS requests.

However, ATC Invalidates are only sent to a port which appears in `list_of_pcie_rc`. It should be uniquely decoded to a single port based on `list_of_ns_sid_high_at_bitpos0`, and ATC Invalidates must be routed to the correct PCIe subsystem to invalidate the cache of ATS Responses in the subsystem. Therefore all TBUs that a PCIe-RC uses must have a unique reverse mapping from stream id to port.



A bad configuration renders the model inactive.

Some configuration can be adjusted by configuration pins. These are only sampled at the negative edge of the reset pin. If you want to use these pins, then you must drive them before sending a negative edge on the reset pin. During `simulation_reset`, the component driving them must also drive this transition again.

The pin `sup_oas` is not supported, instead it is a parameter, as it is assumed that it would be tied to a fixed value in any specific platform.

Debug reads to the registers do not disturb the state.

Writes to registers with Update flags, including debug writes, are ignored if the Update flag is already set to one.

Debug and real accesses to the registers must be 32 bits or 64 bits.

MSIs are issued on the `qtw_pvbus_m` port using attributes that are determined by the parameter `msi_attribute_transform`, while Event queue writes are always issued with `ExtendedID=0`, `UserFlags=0`, `MasterID=0xFFFFffff`. In the hardware, there is no way of distinguishing Event queue writes from MSI writes. However, this provides a mechanism for the model system to distinguish them.

The hardware only has a single cacheability attribute for input transactions, but `PvBus` transports have both inner and outer cacheability.

For non-PCIe-mode TBUs, whose index does not appear in `list_of_pcie_mode` or `list_of_pcie_rc`, for non-cache maintenance operations:

- If the input attribute is any type of device then it is well defined as being outer-shared and Device-nGnRnE or Device-nGnRE. There is no support for Gathering or Reordering.
- If the outer cacheable input attribute is normal, and it is Write-back, this is converted to Inner Write-back Outer Write-back (iWB-oWB) with the desired shareability. No Transient hint is supported and is always treated as non-transient.
- This leaves all other normal memory types that are mapped to Inner Normal Non-cacheable, Outer Normal Non-cacheable outer shared (iNC-oNC-osh).

Therefore, the upstream devices must present the cacheability in the *outer* cacheability attribute on `PvBus` if it is cacheable. If it is a device type then both the inner and outer attributes must be set to the same value. If it is iNC-oNC-osh, then it must be presented as such.

For PCIe-mode TBUs, that is, whose index appears in `list_of_pcie_mode` or `list_of_pcie_rc`:

- Input transactions are from PCIe and the only indication of the memory type is in the NoSnoop bit of the transaction. No shareability is transported. NoSnoop is interpreted as iNC-oNC-osh. ! NoSnoop is interpreted as iWB-oWB-ish (note Inner Shareable).
- If a NoSnoop transaction has an attribute transform applied to it and the result of the transform is weaker than iNC-oNC-osh, it is forced to iNC-oNC-osh. For example, if a NoSnoop transaction uses a page table and is transformed to iWB-oWB-nsh then it is forced to iNC-oNC-osh. However, if the page tables transformed it to a device type, then as all device types are stronger than iNC-oNC-osh, it exits the SMMU as the device type.
- In the model, transactions are classified by their incoming memory attributes as to whether they are NoSnoop or not and then are normalized appropriately:
 - iWB-oWB-any-shareability are interpreted as ! NoSnoop, therefore are normalized to iWB-oWB-ish.

- Anything else is considered NoSnoop, and therefore is normalized to iNC-oNC-osh.
- Translated accesses also have the same interpretation to determine NoSnoop and how they are normalized. Therefore they could enter the system with different attributes to if they entered the SMMU as Untranslated Accesses and were translated by the normal translation process, before exiting downstream of the SMMU and entering the system
- It is expected that there is a component downstream of the SMMU that is aware of the system address map and will override the memory type to `device` for any transaction that accesses a peripheral.

The hardware has a single cacheability on input, and, for transactions that are neither cache-maintenance operations nor PCIe transactions, it normalizes the input to an architectural form before performing the SMMUV3 architectural transform:

- Any device type is left untouched. The input can only represent Device-nGnRE and Device-nGnRnE.
- If the input is Write-back (WB), it is normalized to iWB-oWB with the incoming shareability.
- If the input is anything else, it is normalized to iNC-oNC-osh.

The model accepts full architectural attributes of two levels of cacheability and so has to decide how to interpret this in terms of the hardware. For transactions that are not cache maintenance operations, the model replicates the outer attribute into the inner attribute and then performs the normalization that the hardware does.

The hardware normalizes the architectural output attributes and outputs a single level of cacheability and a user flag (OC) specifying if the architectural attributes were cacheable in the outer cacheable domain. If the transaction is classified as a PCIe transaction, the NoSnoop transform previously described is applied. That is, if the original transaction was NoSnoop then any weaker memory type is strengthened to iNC-oNC-osh, and the following transform is applied:

```

if      iWB-oWB-nsh/ish/osh      then output WB-nsh/ish/osh,  OC = 1
else if i (NC/WT/WB) -o (WB/WT)  then output NC-Sys,      OC = 1
else if i (NC/WT/WB) -oNC        then output NC-Sys,      OC = 0
else if Device-(GRE/nGRE/nGnRE)  then output DV-Sys,      OC = 0
else                             output SO-Sys,          OC = 0

```

The model only normalizes according to PCIe but otherwise leaves the architectural attributes intact on the output bus.



The model deals with groups of transactions with the same attributes and a similar range of addresses. The mapping that is used is stored by the bus infrastructure and is used for subsequent sufficiently similar transactions without needing the intervention of the SMMU model, and therefore are not traced.

Model limitations

- The Performance Monitoring Unit (PMU) has limited functionality. It is intended for demonstration purposes only and does not implement all architecturally required events. It does not implement the `pmusnapshot_ack` or `pmusnapshot_req` interface.

- The model does not implement:
 - RAS.
 - Power control.
 - `sup_oas`, which controls the OAS of the SMMU. This is expected to be constant for a system.
 - The SYSCO interface.
 - The Low-Power Interface.
 - Any of the IMP DEF MPAM registers.
- Cache maintenance operations cannot be inserted into the TBU ports of the SMMU.
- `PVBus` has no representation of the cache stash operations, so they are not supported.
- `TCU_CFG.XLATE_SLOTS` is fixed at 512.
- `TCU_STATUS.GNT_XLATE_SLOTS` always reads as 512.

Security State Determination (SSD)

The model uses the term SSD to mean the security state that the transaction, register, structure, or event belongs to. In the SMMUv3 architecture, the sideband signal `SEC_SID` holds this information for the transactions, but uses a different encoding.

Before RME, `SEC_SID` was a one-bit signal. With RME, in the hardware, it was expanded to two bits. To retain backwards compatibility in the model, `SEC_SID` remains as one-bit in the parameter `howto_identify`, but the second bit can be expressed with `SEC_SID_bit_1`, and its negative logic version `nSEC_SID_bit_1`.

Table 3-1049: Security State Determination (SSD) encoding

Security state	SSD	SEC_SID_bit_1	SEC_SID
Secure	0	0	1
Non-secure	1	0	0
Root (reserved)	2	1	1
Realm	3	1	0

Ports for MMU_600

Table 3-1050: Ports

Name	Protocol	Type	Description
<code>clk_in</code>	<code>ClockSignal</code>	Slave	Clock signal (in RTL <code>ack</code>) This is a clock time-base used by the TCU to spread some of its processing over time, if enabled by the <code>wait_*</code> parameters. The clock must always be connected.
<code>cmd_sync_irpt_ns</code>	<code>Signal</code>	Master	Pulsed interrupt output signal for non-secure <code>CMD_SYNC</code> having a completion signal of <code>SIG_IRQ</code> .
<code>cmd_sync_irpt_s</code>	<code>Signal</code>	Master	Pulsed interrupt output signal for secure <code>CMD_SYNC</code> having a completion signal of <code>SIG_IRQ</code> .
<code>event_q_irpt_ns</code>	<code>Signal</code>	Master	Pulsed interrupt output signal for the non-secure event queue becoming non-empty.

Name	Protocol	Type	Description
event_q_irpt_s	Signal	Master	Pulsed interrupt output signal for the secure event queue becoming non-empty.
evento	Signal	Master	Event signal
global_irpt_ns	Signal	Master	Pulsed interrupt output signal for non-secure SMMU_GERROR(N) signalling an error.
global_irpt_s	Signal	Master	Pulsed interrupt output signal for secure SMMU_S_GERROR(N) signalling an error.
identify	SMMUv3AEMIdentifyProtocol	Master	Map the transaction to the tuple (StreamID, SubStreamID, SubStreamIDValid, SSD) The StreamID that is produced by the implementation of this protocol is not the final StreamID. The final StreamID is produced by using the list_of_ns_sid_high_at_bitpos0/list_of_s_sid_high_at_bitpos0 parameter to map the StreamID based on the upstream port index. Also see the parameter howto_identify which can replace the functionality of this port under certain circumstances.
pri_q_irpt_ns	Signal	Master	Pulsed interrupt output signal for the PRI queue becoming non-empty. Exists only for r1 and higher.
prog_pvbus_s	PVBus	Slave	Register subordinate port (in RTL PROG)
pvbus_id_routed_m[62]	PVBus	Master	This is a special "id-routed" port for transmitting ATC invalidates upstream into the PCIe EndPoints, it is not a normal bus. See the parameter output_id_routed_transform. The FastModels ATC Invalidate and PRI Response protocol specifies how to route and deal with this port.
qtw_pvbus_m	PVBus	Master	Manager port used for Table Walks, MSIs and Queue access. Note that although this is a manager port, it can still receive DVM messages.
sec_override	Signal	Slave	Allow certain registers to be accessible to non-secure accesses from reset, as described in the TCU_SCR register. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sup_btm	Signal	Slave	System supports BTM and will be reflected in the IDR registers. This signal can override the value set by the parameters configuring the IDR registers. If BTM (Broadcast Table Maintenance) is not supported then DVM messages will be ignored. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sup_cohacc	Signal	Slave	System supports COHACC and will be reflected in the IDR registers. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sup_sev	Signal	Slave	System supports SEV and will be reflected in the IDR registers. This signal can override the value set by the parameters configuring the IDR registers. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
tbm_pvbus_m[62]	PVBus	Master	The TBU manager ports that carry transactions that have been translated from the correspondingly numbered tbs_pvbus_s[] port.
tbs_pvbus_s[62]	PVBus	Slave	The TBU subordinate ports that receive transactions to be translated. They will exit the SMMU through the same numbered pvbus_m[] port.

Name	Protocol	Type	Description
tbu_pmu_irpt[62]	Signal	Master	TBU Performance Monitoring Unit interrupt, one per TBU.
tbu_pmusnapshot_ack[62]	Signal	Master	PMU snapshot interface for the TBU, ack a snapshot.
tbu_pmusnapshot_req[62]	Signal	Slave	PMU snapshot interface for the TBU, request a snapshot.
tbu_ras_irpt[62]	Signal	Master	The RAS interrupt pin for errors detected in the TBUs.
tbu_reset_in[62]	Signal	Slave	Reset signals The TBUs can have independent reset signals. Each signal tbu_reset_in[n] corresponds to the TBU using tbs_pvbus_s[n]/tbs_pvbus_s[n] pair. If the SMMU receives a transaction whilst the TBU is expected to be in reset then it will complain using the ArchMsg.Warning.warning trace source. Those tbu_reset_in that correspond to a PCIe-RC connection can be connected to monitor the PCIe-RC's reset signal. If it receives an ATS request when in reset then it will complain in a similar way. You must connect these pins if you wish the TCU_NODE_STATUS for the nodes to be accurate (including any connected to the PCIe-RC).
tcu_pmu_irpt	Signal	Master	TCU Performance Monitoring Unit interrupt
tcu_pmusnapshot_ack	Signal	Master	PMU snapshot interface for the TCU, ack a snapshot.
tcu_pmusnapshot_req	Signal	Slave	PMU snapshot interface This is a four-phase handshake to have the corresponding PMCG perform a capture of the current counter values. PMU snapshot interface for the TCU, request a snapshot.
tcu_ras_irpt	Signal	Master	The RAS interrupt pin for errors detected in the TCU.
tcu_reset_in	Signal	Slave	The reset signal to the TCU interface.

Parameters for MMU_600

TCUCFG_XLATE_SLOTS

Maximum number of outstanding stalled transactions that the SMMU supports Note that TCUCFG_XLATE_SLOTS must be \geq TCUCFG_PTW_SLOTS which currently fixed to 512.
Accepted Values: 512

Type

int

Default value

0x200

all_error_messages_through_trace

Some conditions in the SMMU are so strange that the software programming the SMMU has done something wrong. At this point messages are output to either ArchMsg.Error.* or ArchMsg.Warning.* or to the error stream of the simulator. Outputting to the error stream of the simulator may cause it to return with a non-zero exit status.

If you set this option to true then instead of using the error stream of the simulator it will always use a trace stream allowing the simulation to exit with a zero exit status.

Type

bool

Default value

0x0

behaviour_of_sampled_at_reset_signals

Some configuration signals into the SMMU are sampled on negedge of reset.

However, it can sometimes be hard to arrange to drive a configuration pin before the negedge of reset.

The configuration pins are sampled: 0 -- at negedge reset. 1 -- at negedge reset, but if a later change occurs at the same simulated time, and no transactions have occurred, then they will be resampled and the SMMU reset again.

Type

int

Default value

0x0

cmdq_max_number_of_commands_to_buffer

The command queues can buffer fetched commands before issuing them. This parameter is roughly the maximum number of commands to do this for. The programmer visible effects are that just because the CONS pointer shows a command has been `_consumed_` does not necessarily mean that it has been issued (and completed). Higher values will accentuate this effect.

Type

int

Default value

0xa

enable_device_id_checks

If this parameter is true then the DeviceIDs seen by the GIC are: * for client devices: DeviceID = StreamID + translated_device_id_base * for SMMU-generated MSIs: smmu_msi_device_id

This parameter enables two checks:

* If the DeviceID is used in the `output_attribute_transform` then if it overflows 32 bits then the model will warn. If the DeviceID is not used then it is assumed that the external agent that forms the DeviceID will warn if it overflows.

* If the SMMU supports MSIs, then the model will check that the GIC will be able to distinguish an MSI generated by the SMMU from one generated by a client device.

As the exact mechanism to determine the DeviceID is in the system and not necessarily under control of the SMMU then you can disable these warnings using this parameter.

See also the parameters: `output_attribute_transform` and `msi_attribute_transform`.

Type

bool

Default value

0x1

howto_identify

If 'use-identify' then will use the 'identify' port to determine the SSD, StreamID, SubStreamID. Otherwise, this string extracts them from the transaction's attributes.

Examples:-

```
SEC_SID=ExtendedID[63], SSV=ExtendedID[62], SubstreamID=ExtendedID[51:32],
StreamID=ExtendedID[31:0] nSEC_SID=ExtendedID[63], StreamID=ExtendedID[55:24],
nSSV=ExtendedID[20], SubstreamID=ExtendedID[19:0] StreamID[31:24]=0,
StreamID[23:0]=ExtendedID[23:0], SSV=1[0], ...
```

The StreamID (32 b) is valid if SIDV is 1 or both SIDV and nSIDV are unused.

SEC_SID is 1b in the model. For RME systems, in HW then SEC_SID is 2b, in the model SEC_SID_bit_1 represents bit[1].

The SubstreamID (20 b) is valid if SSV is true.

SIDV == 0 indicates a NoStreamID transaction and the SSD is the PAS of the transaction, SEC_SID is not used.

nSEC_SID, nSSV, nSIDV are available with negative logic. Different attributes are independent and can use negative or positive logic.

Type

string

Default value

```
SEC_SID=ExtendedID[63], SSV=ExtendedID[62], SubstreamID=ExtendedID[51:32],
StreamID=ExtendedID[31:0]
```

list_of_ns_sid_high_at_bitpos0

A comma-separated list of values to bitwise OR into each Non-secure StreamID for each TBU/Node. Bit 0 of the value corresponds to bit 0 of the StreamID.

Each TBU that is connected to a PCIe-RC (see `list_of_pcie_rc`) must serve a unique contiguous subset of StreamIDs as determined by their top bits. This is used in order to know which port to route ATC Invalidates and PRI Responses to the PCIe subsystems.

The empty string corresponds to all 0s.

Type

string

Default value**list_of_pcie_mode**

A comma-separated list of ranges of ports that represent TBUs that are attached to PCIe Root Complexes (PCIe-RC). A single PCIe-RC might use several TBUs and stripe accesses across them. The attributes handling for these TBUs are slightly different in that if the PCIe transaction is NoSnoop and the output attributes of the translation would be weaker than iNC-oNC-osh then the output is forced to iNC-oNC-osh.

iNC-oNC-osh == "inner normal non-cacheable, out normal non-cacheable, outer shared"

Type

string

Default value**list_of_pcie_rc**

This is a list of ports that are connected to PCIe Root Complex (PCIe-RC) by a protocol called DTI-ATS. This port is used to transport ATS and PRI Requests to the SMMU from the PCIe-RC.

In the real hardware, then the PCIe-RC uses this port for ATS/PRI, and then the actual transactions go through separate TBUs. In the model, then this port can accept actual transactions as well. However, in the model, then the ATC Invalidates and the PRI Responses need to be transferred over the corresponding pvbush_id_routed_m port as DTI-ATS is bidirectional, but PVBush is not.

Type

string

Default value**list_of_s_sid_high_at_bitpos0**

A comma-separated list of values to bitwise OR into each Secure StreamID for each TBU/Node. Bit 0 of the value corresponds to bit 0 of the StreamID.

The empty string corresponds to all 0s.

"use-ns" can be used to apply the list_of_ns_sid_high_at_bitpos0 values instead.

Type

string

Default value

msi_attribute_transform

Transform downstream attributes of MSI transactions. * ""/"none" -- no transform * How to alter output attributes of SMMU-generated MSIs. Example: "UserFlags[15:0]=smmu_msi_device_id[31:16], MasterID[15:0]=smmu_msi_device_id[15:0], ExtendedID=0"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags

RHS Symbols: * the parameter smmu_msi_device_id * 'interrupt_kind' * 0/1 -- EVENTQ s/ns * 2 -- PRIQ * 3/4 -- CMD_SYNC s/ns * 5/6 -- GERROR s/ns * 7/8 -- PMCG s/ns * 9/10/11 -- RAS FHI/ERI/CRI * 12/13 -- gpf_far/gpt_cfg_far * 14/15/16/17 -- Realm EVENTQ/PRIQ/CMDQ/GERROR * HWATTR_KIND_0: PBHA information * numeric literals.

ExtendedID/MasterID/UserFlags start with values {0, 0xFFFFFFFF, 0} respectively.

Any bits with no transform are unchanged.

This transform can be used to determine the DeviceID passed to the GIC to distinguish MSIs generated by the SMMU from those generated by client devices.

NOTE: * see also 'output_attribute_transform' and enable_device_id_checks.

Type

string

Default value

ExtendedID[31:0]=smmu_msi_device_id, MasterID=0xFFFFFFFF

number_of_ports

The number of port pairs that the SMMU has.

Type

int

Default value

0x1

output_attribute_transform

Transform the downstream attributes of a translated transaction.

* "" or "none" -- the input and output attributes are identical.

* How to alter the output attributes, e.g. "ExtendedID[15:0]=DeviceID[15:0], UserFlags[31]=nSSV, UserFlags[19:0]=SubstreamID"

The attributes that can appear on the left hand side of the transform are ExtendedID, MasterID and UserFlags.

The source attributes that can be used are: * ExtendedID/MasterID/UserFlags -- the incoming attributes. * DeviceID -- StreamID + translated_device_id_base * StreamID/

SubstreamID/SSV/SEC_SID * nSEC_SID/nSSV -- the negative logic versions. * St1PBHA/St2PBHA -- the Page Based Hardware Attributes from any used leaf descriptors (or zero if not used). * STE_IMPDEF1 -- STE[127:116]

The right hand side may also contain numeric literals. Any bits of the attributes that have no transform specified are retained from the input.

The StreamID has had ns_sid_high/s_sid_high ORred into it for the appropriate TBU.

Type

string

Default value

ExtendedID[31:0]=DeviceID

output_id_routed_transform

The SMMU generates the following ID-routed transaction on the pvbus_id_routed_m bus: * ATC Invalidate * PRI Response

This parameter expresses how the SMMU should express: * the StreamID * the Trusted (T) bit

The value is a comma-separated list of assignments:

Address[27:12]=StreamID[15:0], ExtendedID[60]=T, ExtendedID[15:0]=StreamID[31:16]

Address bits[11:0] cannot be used.

The LHS can be one of: * PAS * MasterID/ExtendedID/UserFlags * Address

The RHS can be one of: * a numeric constant * SSD * T or negative version nT * StreamID

For realm (or 'Trusted') transactions, then SSD=0b11, T=1, nT=0. For non-secure (or 'Non-Trusted') transactions, then SSD=0b01, T=0, nT=1

Type

string

Default value

Address[27:12]=StreamID[15:0], PAS=SSD

prefetch_only_requests

The simulator supports 'prefetch-only' DMI requests, which can occur for anytime for any reason and are intended to be invisible to the end execution of the model and to the user.

0 -- deny all prefetch-only requests 1 -- use debug requests for any page table walks -- form and use debug TLB/cache entries -- any faults will not record, but deny the prefetch request 2 -- treat prefetch-only requests like normal transactions -- use normal page table walk transactions -- use and form normal TLB/cache entries -- faults will alter the programmer visible state of the SMMU

0 is the safest.

1 treats the access like a debug request and requires that debug page table walks are treated correctly downstream. Any descriptors that need HTTU to allow the transaction to proceed will fail the request.

2 is dangerous, it use real transactions and reports faults that are unphysical. Real transactions can be wait()ed and this disobeys the SystemC spec for get_direct_mem_ptr().

Type

int

Default value

0x0

sec_override

The IMP DEF port sec_override controls whether some of the registers are accessible to secure or non-secure transactions. This parameter is the default value assumed for that port if the port is not driven by a signal.

Type

bool

Default value

0x0

seed

Used to seed the pseudo-random number generator that the SMMU model uses.

Type

int

Default value

0x12345678

size_of_cd_cache

The number of entries in the cache holding CD structures. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_l1cd_cache

The number of entries in the cache holding L1CD descriptors. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_l1ste_cache

The number of entries in the cache holding L1STE descriptors. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_ste_cache

The number of entries in the cache holding STE structures. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_tlb

The number of entries in the TLB. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

smmu_msi_device_id

When appropriately enabled, assume that MSIs that are generated by the SMMU are presented to the GIC with this DeviceID.

See parameter msi_attribute_transform and enable_device_id_checks.

Type

int

Default value

0x0

sup_btm

The default value of the register field SMMU_IDR0.BTM and SMMU_ROOT_IDR0.BGPTM (when supported). This enables Broadcast TLB maintenance.

Type

bool

Default value

0x1

sup_cohacc

The default value of the register SMMU_IDR0.COHAACC

Type

bool

Default value

0x1

sup_oas

The hardware has an input port sup_oas[2:0] that indicates what output address size (OAS) the _system_ has. This is sampled at reset.

The model does not have this port as it is expected to be a constant for the system and not to change. Instead it is just a parameter.

The allowed values are: 0 -- 32 bits 1 -- 36 bits 2 -- 40 bits 3 -- 42 bits 4 -- 44 bits 5 -- 48 bits

Type

int

Default value

0x5

sup_sev

The default value of the register SMMU_IDR0.SEV

Type

bool

Default value

0x1

tlb_when_do_f_tlb_conflict_on_overlap

If a TLB entry is created by a walk and it overlaps an existing entry. Then there are some architectural situations where the result is known. For all others, then an implementation is allowed to use an UNPREDICTABLE combination of the two entries, or it can generate F_TLB_CONFLICT:

0 -- never generate 1 -- sometimes generate 2 -- always generate

Conflicts between global and non-global entries are not detected by the model.

Type

int

Default value

0x0

translated_device_id_base

When appropriately enabled, assume that client device accesses are translated to a DeviceID as seen by the GIC of: StreamID + translated_device_id_base

See parameter output_attribute_transform and enable_device_id_checks.

Type

int

Default value

0x0

tw_qs_attribute_transform

Transform downstream attributes of table walk and queue transactions.

* ""/"none" -- no transform * How to alter the output attributes. Example:

"ExtendedID[35:32]=HWATTR_KIND_0"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags

RHS Symbols: * HWATTR_KIND_0: PBHA information * kind * for a read: * 0/1 -- L1STE/STE * 2/3 -- L1CD/CD * 4/5 -- S1/S2 TTD (including CAS) * 6 -- CMDQ * 7 -- VMS * 11/12 -- LOGPT/L1GPT * 13/14 -- LODPT/L1DPT * for a write * 0 -- EVENTQ * 1 -- PRIQ * numeric literals.

ExtendedID/MasterID/UserFlags start with values {0, 0xFFFFfff, 0} respectively.

Any bits with no transform are unchanged.

NOTE: * see also 'output_attribute_transform' and 'msi_attribute_transform'.

Type

string

Default value

version

The version of this product

Type

string

Default value

r0p0

wait_cmdq_ticks

This is the time to wait before doing something on the command queue. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_eventq_ticks

This is the time to wait before doing something on the event queue. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_imp_def_work_ticks

This is the time to wait before doing an IMP DEF operation. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

The IMP DEF work in this case is the number of ticks between raising pmusnapshot_req and pmusnapshot_ack being raised, and the converse operation.

Type

int

Default value

0x0

wait_misc_async_actions_ticks

This is the time to wait before doing an async action that could be delayed is performed. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_msi_ticks

This is the time to wait before sending an MSI. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_pri_req_ticks

This is the time to wait before processing a PRI Request. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_pri_resp_ticks

This is the time to wait before sending a PRI Response back to the PCIe subsystem. When a PRI Response is an auto-response then the ATC might immediately make a new ATS request, that immediately fails, that immediately makes a PRI Request, that auto-responds, etc. To break this loop, then we introduce a minimum time on all PRI Responses to give other components in the system a chance to run. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x1

3.10.55 MMU_700

This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1051: IP revisions support

Revision	Quality level
r0p0	Full support
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- sup_btm
- sup_httu
- sup_sev

Iris and MTI instances for MMU_700

This model has the following Iris instances:

Table 3-1052: MMU_700 Iris instances

InstanceName	ComponentName
MMU_700	MMU_700
MMU_700.register_file[0]	PVBusSlave
MMU_700.service_request_tbu[0]	PVBusSlave
MMU_700.service_request_tbu[10]	PVBusSlave
MMU_700.service_request_tbu[11]	PVBusSlave
MMU_700.service_request_tbu[12]	PVBusSlave
MMU_700.service_request_tbu[13]	PVBusSlave
MMU_700.service_request_tbu[14]	PVBusSlave
MMU_700.service_request_tbu[15]	PVBusSlave
MMU_700.service_request_tbu[16]	PVBusSlave
MMU_700.service_request_tbu[17]	PVBusSlave
MMU_700.service_request_tbu[18]	PVBusSlave
MMU_700.service_request_tbu[19]	PVBusSlave
MMU_700.service_request_tbu[1]	PVBusSlave
MMU_700.service_request_tbu[20]	PVBusSlave
MMU_700.service_request_tbu[21]	PVBusSlave
MMU_700.service_request_tbu[22]	PVBusSlave
MMU_700.service_request_tbu[23]	PVBusSlave
MMU_700.service_request_tbu[24]	PVBusSlave
MMU_700.service_request_tbu[25]	PVBusSlave
MMU_700.service_request_tbu[26]	PVBusSlave
MMU_700.service_request_tbu[27]	PVBusSlave
MMU_700.service_request_tbu[28]	PVBusSlave
MMU_700.service_request_tbu[29]	PVBusSlave
MMU_700.service_request_tbu[2]	PVBusSlave
MMU_700.service_request_tbu[30]	PVBusSlave
MMU_700.service_request_tbu[31]	PVBusSlave
MMU_700.service_request_tbu[32]	PVBusSlave
MMU_700.service_request_tbu[33]	PVBusSlave
MMU_700.service_request_tbu[34]	PVBusSlave
MMU_700.service_request_tbu[35]	PVBusSlave

InstanceName	ComponentName
MMU_700.service_request_tbu[36]	PVBusSlave
MMU_700.service_request_tbu[37]	PVBusSlave
MMU_700.service_request_tbu[38]	PVBusSlave
MMU_700.service_request_tbu[39]	PVBusSlave
MMU_700.service_request_tbu[3]	PVBusSlave
MMU_700.service_request_tbu[40]	PVBusSlave
MMU_700.service_request_tbu[41]	PVBusSlave
MMU_700.service_request_tbu[42]	PVBusSlave
MMU_700.service_request_tbu[43]	PVBusSlave
MMU_700.service_request_tbu[44]	PVBusSlave
MMU_700.service_request_tbu[45]	PVBusSlave
MMU_700.service_request_tbu[46]	PVBusSlave
MMU_700.service_request_tbu[47]	PVBusSlave
MMU_700.service_request_tbu[48]	PVBusSlave
MMU_700.service_request_tbu[49]	PVBusSlave
MMU_700.service_request_tbu[4]	PVBusSlave
MMU_700.service_request_tbu[50]	PVBusSlave
MMU_700.service_request_tbu[51]	PVBusSlave
MMU_700.service_request_tbu[52]	PVBusSlave
MMU_700.service_request_tbu[53]	PVBusSlave
MMU_700.service_request_tbu[54]	PVBusSlave
MMU_700.service_request_tbu[55]	PVBusSlave
MMU_700.service_request_tbu[56]	PVBusSlave
MMU_700.service_request_tbu[57]	PVBusSlave
MMU_700.service_request_tbu[58]	PVBusSlave
MMU_700.service_request_tbu[59]	PVBusSlave
MMU_700.service_request_tbu[5]	PVBusSlave
MMU_700.service_request_tbu[60]	PVBusSlave
MMU_700.service_request_tbu[61]	PVBusSlave
MMU_700.service_request_tbu[62]	PVBusSlave
MMU_700.service_request_tbu[63]	PVBusSlave
MMU_700.service_request_tbu[6]	PVBusSlave
MMU_700.service_request_tbu[7]	PVBusSlave
MMU_700.service_request_tbu[8]	PVBusSlave
MMU_700.service_request_tbu[9]	PVBusSlave
MMU_700.tbu[0]	PVBusMapper
MMU_700.tbu[10]	PVBusMapper
MMU_700.tbu[11]	PVBusMapper
MMU_700.tbu[12]	PVBusMapper

InstanceName	ComponentName
MMU_700.tbu[13]	PVBusMapper
MMU_700.tbu[14]	PVBusMapper
MMU_700.tbu[15]	PVBusMapper
MMU_700.tbu[16]	PVBusMapper
MMU_700.tbu[17]	PVBusMapper
MMU_700.tbu[18]	PVBusMapper
MMU_700.tbu[19]	PVBusMapper
MMU_700.tbu[1]	PVBusMapper
MMU_700.tbu[20]	PVBusMapper
MMU_700.tbu[21]	PVBusMapper
MMU_700.tbu[22]	PVBusMapper
MMU_700.tbu[23]	PVBusMapper
MMU_700.tbu[24]	PVBusMapper
MMU_700.tbu[25]	PVBusMapper
MMU_700.tbu[26]	PVBusMapper
MMU_700.tbu[27]	PVBusMapper
MMU_700.tbu[28]	PVBusMapper
MMU_700.tbu[29]	PVBusMapper
MMU_700.tbu[2]	PVBusMapper
MMU_700.tbu[30]	PVBusMapper
MMU_700.tbu[31]	PVBusMapper
MMU_700.tbu[32]	PVBusMapper
MMU_700.tbu[33]	PVBusMapper
MMU_700.tbu[34]	PVBusMapper
MMU_700.tbu[35]	PVBusMapper
MMU_700.tbu[36]	PVBusMapper
MMU_700.tbu[37]	PVBusMapper
MMU_700.tbu[38]	PVBusMapper
MMU_700.tbu[39]	PVBusMapper
MMU_700.tbu[3]	PVBusMapper
MMU_700.tbu[40]	PVBusMapper
MMU_700.tbu[41]	PVBusMapper
MMU_700.tbu[42]	PVBusMapper
MMU_700.tbu[43]	PVBusMapper
MMU_700.tbu[44]	PVBusMapper
MMU_700.tbu[45]	PVBusMapper
MMU_700.tbu[46]	PVBusMapper
MMU_700.tbu[47]	PVBusMapper
MMU_700.tbu[48]	PVBusMapper

InstanceName	ComponentName
MMU_700.tb[49]	PVBusMapper
MMU_700.tb[4]	PVBusMapper
MMU_700.tb[50]	PVBusMapper
MMU_700.tb[51]	PVBusMapper
MMU_700.tb[52]	PVBusMapper
MMU_700.tb[53]	PVBusMapper
MMU_700.tb[54]	PVBusMapper
MMU_700.tb[55]	PVBusMapper
MMU_700.tb[56]	PVBusMapper
MMU_700.tb[57]	PVBusMapper
MMU_700.tb[58]	PVBusMapper
MMU_700.tb[59]	PVBusMapper
MMU_700.tb[5]	PVBusMapper
MMU_700.tb[60]	PVBusMapper
MMU_700.tb[61]	PVBusMapper
MMU_700.tb[62]	PVBusMapper
MMU_700.tb[63]	PVBusMapper
MMU_700.tb[6]	PVBusMapper
MMU_700.tb[7]	PVBusMapper
MMU_700.tb[8]	PVBusMapper
MMU_700.tb[9]	PVBusMapper

This model has the following MTI trace components:

Table 3-1053: MMU_700 MTI instances

InstanceName	ComponentName
MMU_700	MMU_700
MMU_700.register_file[0]	PVBusSlave
MMU_700.service_request_tbu[0]	PVBusSlave
MMU_700.service_request_tbu[10]	PVBusSlave
MMU_700.service_request_tbu[11]	PVBusSlave
MMU_700.service_request_tbu[12]	PVBusSlave
MMU_700.service_request_tbu[13]	PVBusSlave
MMU_700.service_request_tbu[14]	PVBusSlave
MMU_700.service_request_tbu[15]	PVBusSlave
MMU_700.service_request_tbu[16]	PVBusSlave
MMU_700.service_request_tbu[17]	PVBusSlave
MMU_700.service_request_tbu[18]	PVBusSlave
MMU_700.service_request_tbu[19]	PVBusSlave
MMU_700.service_request_tbu[1]	PVBusSlave

InstanceName	ComponentName
MMU_700.service_request_tbu[20]	PVBusSlave
MMU_700.service_request_tbu[21]	PVBusSlave
MMU_700.service_request_tbu[22]	PVBusSlave
MMU_700.service_request_tbu[23]	PVBusSlave
MMU_700.service_request_tbu[24]	PVBusSlave
MMU_700.service_request_tbu[25]	PVBusSlave
MMU_700.service_request_tbu[26]	PVBusSlave
MMU_700.service_request_tbu[27]	PVBusSlave
MMU_700.service_request_tbu[28]	PVBusSlave
MMU_700.service_request_tbu[29]	PVBusSlave
MMU_700.service_request_tbu[2]	PVBusSlave
MMU_700.service_request_tbu[30]	PVBusSlave
MMU_700.service_request_tbu[31]	PVBusSlave
MMU_700.service_request_tbu[32]	PVBusSlave
MMU_700.service_request_tbu[33]	PVBusSlave
MMU_700.service_request_tbu[34]	PVBusSlave
MMU_700.service_request_tbu[35]	PVBusSlave
MMU_700.service_request_tbu[36]	PVBusSlave
MMU_700.service_request_tbu[37]	PVBusSlave
MMU_700.service_request_tbu[38]	PVBusSlave
MMU_700.service_request_tbu[39]	PVBusSlave
MMU_700.service_request_tbu[3]	PVBusSlave
MMU_700.service_request_tbu[40]	PVBusSlave
MMU_700.service_request_tbu[41]	PVBusSlave
MMU_700.service_request_tbu[42]	PVBusSlave
MMU_700.service_request_tbu[43]	PVBusSlave
MMU_700.service_request_tbu[44]	PVBusSlave
MMU_700.service_request_tbu[45]	PVBusSlave
MMU_700.service_request_tbu[46]	PVBusSlave
MMU_700.service_request_tbu[47]	PVBusSlave
MMU_700.service_request_tbu[48]	PVBusSlave
MMU_700.service_request_tbu[49]	PVBusSlave
MMU_700.service_request_tbu[4]	PVBusSlave
MMU_700.service_request_tbu[50]	PVBusSlave
MMU_700.service_request_tbu[51]	PVBusSlave
MMU_700.service_request_tbu[52]	PVBusSlave
MMU_700.service_request_tbu[53]	PVBusSlave
MMU_700.service_request_tbu[54]	PVBusSlave
MMU_700.service_request_tbu[55]	PVBusSlave

InstanceName	ComponentName
MMU_700.service_request_tbu[56]	PVBusSlave
MMU_700.service_request_tbu[57]	PVBusSlave
MMU_700.service_request_tbu[58]	PVBusSlave
MMU_700.service_request_tbu[59]	PVBusSlave
MMU_700.service_request_tbu[5]	PVBusSlave
MMU_700.service_request_tbu[60]	PVBusSlave
MMU_700.service_request_tbu[61]	PVBusSlave
MMU_700.service_request_tbu[62]	PVBusSlave
MMU_700.service_request_tbu[63]	PVBusSlave
MMU_700.service_request_tbu[6]	PVBusSlave
MMU_700.service_request_tbu[7]	PVBusSlave
MMU_700.service_request_tbu[8]	PVBusSlave
MMU_700.service_request_tbu[9]	PVBusSlave
MMU_700.tbu[0]	PVBusMapper
MMU_700.tbu[10]	PVBusMapper
MMU_700.tbu[11]	PVBusMapper
MMU_700.tbu[12]	PVBusMapper
MMU_700.tbu[13]	PVBusMapper
MMU_700.tbu[14]	PVBusMapper
MMU_700.tbu[15]	PVBusMapper
MMU_700.tbu[16]	PVBusMapper
MMU_700.tbu[17]	PVBusMapper
MMU_700.tbu[18]	PVBusMapper
MMU_700.tbu[19]	PVBusMapper
MMU_700.tbu[1]	PVBusMapper
MMU_700.tbu[20]	PVBusMapper
MMU_700.tbu[21]	PVBusMapper
MMU_700.tbu[22]	PVBusMapper
MMU_700.tbu[23]	PVBusMapper
MMU_700.tbu[24]	PVBusMapper
MMU_700.tbu[25]	PVBusMapper
MMU_700.tbu[26]	PVBusMapper
MMU_700.tbu[27]	PVBusMapper
MMU_700.tbu[28]	PVBusMapper
MMU_700.tbu[29]	PVBusMapper
MMU_700.tbu[2]	PVBusMapper
MMU_700.tbu[30]	PVBusMapper
MMU_700.tbu[31]	PVBusMapper
MMU_700.tbu[32]	PVBusMapper

InstanceName	ComponentName
MMU_700.tbu[33]	PVBusMapper
MMU_700.tbu[34]	PVBusMapper
MMU_700.tbu[35]	PVBusMapper
MMU_700.tbu[36]	PVBusMapper
MMU_700.tbu[37]	PVBusMapper
MMU_700.tbu[38]	PVBusMapper
MMU_700.tbu[39]	PVBusMapper
MMU_700.tbu[3]	PVBusMapper
MMU_700.tbu[40]	PVBusMapper
MMU_700.tbu[41]	PVBusMapper
MMU_700.tbu[42]	PVBusMapper
MMU_700.tbu[43]	PVBusMapper
MMU_700.tbu[44]	PVBusMapper
MMU_700.tbu[45]	PVBusMapper
MMU_700.tbu[46]	PVBusMapper
MMU_700.tbu[47]	PVBusMapper
MMU_700.tbu[48]	PVBusMapper
MMU_700.tbu[49]	PVBusMapper
MMU_700.tbu[4]	PVBusMapper
MMU_700.tbu[50]	PVBusMapper
MMU_700.tbu[51]	PVBusMapper
MMU_700.tbu[52]	PVBusMapper
MMU_700.tbu[53]	PVBusMapper
MMU_700.tbu[54]	PVBusMapper
MMU_700.tbu[55]	PVBusMapper
MMU_700.tbu[56]	PVBusMapper
MMU_700.tbu[57]	PVBusMapper
MMU_700.tbu[58]	PVBusMapper
MMU_700.tbu[59]	PVBusMapper
MMU_700.tbu[5]	PVBusMapper
MMU_700.tbu[60]	PVBusMapper
MMU_700.tbu[61]	PVBusMapper
MMU_700.tbu[62]	PVBusMapper
MMU_700.tbu[63]	PVBusMapper
MMU_700.tbu[6]	PVBusMapper
MMU_700.tbu[7]	PVBusMapper
MMU_700.tbu[8]	PVBusMapper
MMU_700.tbu[9]	PVBusMapper

MMU_700 contains the following CADI targets:

- MMU_700

About MMU_700

MMU_600 and MMU_700 are SMMUv3-compliant devices and are used for I/O virtualization of devices.

The hardware is a distributed SMMU. It consists of the following:

- A single Translation Control Unit (TCU), which has a port for the programming interface of the SMMU, receives DVM messages, and does all the page walking and queue manipulation, for example.
- One or more Translation Bus Units (TBUs), which translate transactions from upstream client devices into downstream transactions.
- Zero or more connections to PCIe Root Complexes (PCIe-RCs). A maximum of 62 TBUs and PCIe-RCs can be attached.
- An interconnect that connects the TBUs and PCIe-RCs to the TCU.

A PCIe-RC has one connection to the TCU to make ATS requests but the PCIe-RC uses one or more TBUs to transform the transactions and pass them to the memory system. In the model, the TBUs used are listed in the parameter `list_of_pcie_mode`. The SMMU does not know which TBUs a particular PCIe-RC is attached to.

The TBU ORs a value into each StreamID that it receives. In the model, this is configured by the following parameters:

- `list_of_s_sid_high_at_bitpos0`
- `list_of_ns_sid_high_at_bitpos0`

The TCU, TBU, and the interconnect are all represented by this single model component.

In the model, the `tbs_pvbuss_s[i]` and `tbm_pvbuss_m[i]` port pair represent a TBU *i*, or `tbs_pvbuss_s[i]` represents an incoming connection for a PCIe-RC. The corresponding reverse connection from the TCU to the PCIe-RC is by a special bus called `pvbuss_id_routed_m` that is used to transport ATC Invalidates to the PCIe-RC.

To reduce system construction complexity, the `tbs_pvbuss_s[i]` and `tbm_pvbuss_m[i]` pair also acts as a TBU so that the PCIe-RC does not need to separate its normal transactions and its ATS requests.

However, ATC Invalidates are only sent to a port which appears in `list_of_pcie_rc`. It should be uniquely decoded to a single port based on `list_of_ns_sid_high_at_bitpos0`, and ATC Invalidates must be routed to the correct PCIe subsystem to invalidate the cache of ATS Responses in the subsystem. Therefore all TBUs that a PCIe-RC uses must have a unique reverse mapping from stream id to port.



A bad configuration renders the model inactive.

Some configuration can be adjusted by configuration pins. These are only sampled at the negative edge of the reset pin. If you want to use these pins, then you must drive them before sending a negative edge on the reset pin. During `simulation_reset`, the component driving them must also drive this transition again.

The pin `sup_oas` is not supported, instead it is a parameter, as it is assumed that it would be tied to a fixed value in any specific platform.

Debug reads to the registers do not disturb the state.

Writes to registers with Update flags, including debug writes, are ignored if the Update flag is already set to one.

Debug and real accesses to the registers must be 32 bits or 64 bits.

MSIs are issued on the `qtw_pvbuss_m` port using attributes that are determined by the parameter `msi_attribute_transform`, while Event queue writes are always issued with `ExtendedID=0`, `UserFlags=0`, `MasterID=0xFFFFFFFF`. In the hardware, there is no way of distinguishing Event queue writes from MSI writes. However, this provides a mechanism for the model system to distinguish them.

The hardware only has a single cacheability attribute for input transactions, but `PVBUS` transports have both inner and outer cacheability.

For non-PCIe-mode TBUs, whose index does not appear in `list_of_pcie_mode` OR `list_of_pcie_rc`, for non-cache maintenance operations:

- If the input attribute is any type of device then it is well defined as being outer-shared and Device-nGnRnE or Device-nGnRE. There is no support for Gathering or Reordering.
- If the outer cacheable input attribute is normal, and it is Write-back, this is converted to Inner Write-back Outer Write-back (iWB-oWB) with the desired shareability. No Transient hint is supported and is always treated as non-transient.
- This leaves all other normal memory types that are mapped to Inner Normal Non-cacheable, Outer Normal Non-cacheable outer shared (iNC-oNC-osh).

Therefore, the upstream devices must present the cacheability in the *outer* cacheability attribute on `PVBUS` if it is cacheable. If it is a device type then both the inner and outer attributes must be set to the same value. If it is iNC-oNC-osh, then it must be presented as such.

For PCIe-mode TBUs, that is, whose index appears in `list_of_pcie_mode` OR `list_of_pcie_rc`:

- Input transactions are from PCIe and the only indication of the memory type is in the NoSnoop bit of the transaction. No shareability is transported. NoSnoop is interpreted as iNC-oNC-osh. ! NoSnoop is interpreted as iWB-oWB-ish (note Inner Shareable).

- If a NoSnoop transaction has an attribute transform applied to it and the result of the transform is weaker than iNC-oNC-osh, it is forced to iNC-oNC-osh. For example, if a NoSnoop transaction uses a page table and is transformed to iWB-oWB-nsh then it is forced to iNC-oNC-osh. However, if the page tables transformed it to a device type, then as all device types are stronger than iNC-oNC-osh, it exits the SMMU as the device type.
- In the model, transactions are classified by their incoming memory attributes as to whether they are NoSnoop or not and then are normalized appropriately:
 - iWB-oWB-any-shareability are interpreted as ! NoSnoop, therefore are normalized to iWB-oWB-ish.
 - Anything else is considered NoSnoop, and therefore is normalized to iNC-oNC-osh.
- Translated accesses also have the same interpretation to determine NoSnoop and how they are normalized. Therefore they could enter the system with different attributes to if they entered the SMMU as Untranslated Accesses and were translated by the normal translation process, before exiting downstream of the SMMU and entering the system
- It is expected that there is a component downstream of the SMMU that is aware of the system address map and will override the memory type to device for any transaction that accesses a peripheral.

The hardware has a single cacheability on input, and, for transactions that are neither cache-maintenance operations nor PCIe transactions, it normalizes the input to an architectural form before performing the SMMUV3 architectural transform:

- Any device type is left untouched. The input can only represent Device-nGnRE and Device-nGnRnE.
- If the input is Write-back (WB), it is normalized to iWB-oWB with the incoming shareability.
- If the input is anything else, it is normalized to iNC-oNC-osh.

The model accepts full architectural attributes of two levels of cacheability and so has to decide how to interpret this in terms of the hardware. For transactions that are not cache maintenance operations, the model replicates the outer attribute into the inner attribute and then performs the normalization that the hardware does.

The hardware normalizes the architectural output attributes and outputs a single level of cacheability and a user flag (OC) specifying if the architectural attributes were cacheable in the outer cacheable domain. If the transaction is classified as a PCIe transaction, the NoSnoop transform previously described is applied. That is, if the original transaction was NoSnoop then any weaker memory type is strengthened to iNC-oNC-osh, and the following transform is applied:

```

if      iWB-oWB-nsh/ish/osh      then output WB-nsh/ish/osh, OC = 1
else if i (NC/WT/WB) -o (WB/WT) then output NC-Sys,          OC = 1
else if i (NC/WT/WB) -oNC        then output NC-Sys,          OC = 0
else if Device-(GRE/nGRE/nGnRE) then output DV-Sys,          OC = 0
else                               output SO-Sys,            OC = 0

```

The model only normalizes according to PCIe but otherwise leaves the architectural attributes intact on the output bus.



The model deals with groups of transactions with the same attributes and a similar range of addresses. The mapping that is used is stored by the bus infrastructure and is used for subsequent sufficiently similar transactions without needing the intervention of the SMMU model, and therefore are not traced.

Model limitations

- The Performance Monitoring Unit (PMU) has limited functionality. It is intended for demonstration purposes only and does not implement all architecturally required events. It does not implement the `pmusnapshot_ack` or `pmusnapshot_req` interface.
- The model does not implement:
 - RAS.
 - Power control.
 - `sup_oas`, which controls the OAS of the SMMU. This is expected to be constant for a system.
 - The SYSCO interface.
 - The Low-Power Interface.
 - Any of the IMP DEF MPAM registers.
 - The following registers in MMU-700 r1p0:
 - `TCU_SYSDISC{2-17}`
 - `TBU_SYSDISC{4-14}`
- Cache maintenance operations cannot be inserted into the TBU ports of the SMMU.
- `PVBus` has no representation of the cache stash operations, so they are not supported.
- `TCU_CFG.XLATE_SLOTS` is fixed at 512.
- `TCU_STATUS.GNT_XLATE_SLOTS` always reads as 512.
- The `HWATTR` side-band signal is available by configuring `output_attribute_transform`, `msi_attribute_transform`, and `tw_qs_attribute_transform` with `HWATTR_KIND_0`. For some transactions, the `HWATTR` comes from the `SMMU_S_AGBPA[3:0]`. In the hardware, this register has an Update bit[31] that should be written as 1 and is turned to zero when all transactions using the old value have completed. The model does not implement this behavior and the Update bit is RAZ/WI.

Security State Determination (SSD)

The model uses the term SSD to mean the security state that the transaction, register, structure, or event belongs to. In the SMMUv3 architecture, the sideband signal `SEC_SID` holds this information for the transactions, but uses a different encoding.

Before RME, `SEC_SID` was a one-bit signal. With RME, in the hardware, it was expanded to two bits. To retain backwards compatibility in the model, `SEC_SID` remains as one-bit in the parameter `howto_identify`, but the second bit can be expressed with `SEC_SID_bit_1`, and its negative logic version `nSEC_SID_bit_1`.

Table 3-1054: Security State Determination (SSD) encoding

Security state	SSD	SEC_SID_bit_1	SEC_SID
Secure	0	0	1
Non-secure	1	0	0
Root (reserved)	2	1	1
Realm	3	1	0

Ports for MMU_700**Table 3-1055: Ports**

Name	Protocol	Type	Description
axi_stream_msi_addr_to_match_s	Value_64	Slave	Any SMMU-originated MSI which exactly matches the address in this port will be sent through the AXI stream port axi_stream_msi_m which is usually connected to the GIC through axi_stream_msi_s. As MSIs are 32 bit aligned then if bits[1:0] != 0 or the address is above OAS then this is effectively disabled. NOTE that the model does not support tcu_sid[31:0] which is the MSI DeviceID to send on axi_stream_msi_m. Instead the parameter smmu_msi_device_id is used. See also the parameters: axi_stream_msi_TID and axi_stream_msi_TDEST. The default value of this port is set by the parameter axi_stream_msi_addr_to_match. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
axi_stream_msi_m	PVBus	Master	Manager port used for sending SMMU originated MSIs directly to the GIC
clk_in	ClockSignal	Slave	Clock signal (in RTL aclk) This is a clock time-base used by the TCU to spread some of its processing over time, if enabled by the wait_* parameters. The clock must always be connected.
cmd_sync_irpt_ns	Signal	Master	Pulsed interrupt output signal for non-secure CMD_SYNC having a completion signal of SIG_IRQ.
cmd_sync_irpt_s	Signal	Master	Pulsed interrupt output signal for secure CMD_SYNC having a completion signal of SIG_IRQ.
event_q_irpt_ns	Signal	Master	Pulsed interrupt output signal for the non-secure event queue becoming non-empty.
event_q_irpt_s	Signal	Master	Pulsed interrupt output signal for the secure event queue becoming non-empty.
evento	Signal	Master	Event signal
global_irpt_ns	Signal	Master	Pulsed interrupt output signal for non-secure SMMU_GERROR(N) signalling an error.
global_irpt_s	Signal	Master	Pulsed interrupt output signal for secure SMMU_S_GERROR(N) signalling an error.
identify	SMMUv3AEMIdentifyProtocol	Master	Map the transaction to the tuple (StreamID, SubStreamID, SubStreamIDValid, SSD) The StreamID that is produced by the implementation of this protocol is not the final StreamID. The final StreamID is produced by using the list_of_ns_sid_high_at_bitpos0/list_of_s_sid_high_at_bitpos0 parameter to map the StreamID based on the upstream port index. Also see the parameter howto_identify which can replace the functionality of this port under certain circumstances.

Name	Protocol	Type	Description
pri_q_irpt_ns	Signal	Master	Pulsed interrupt output signal for the PRI queue becoming non-empty.
prog_pvbus_s	PVBus	Slave	Register subordinate port (in RTL PROG)
pvbus_id_routed_m[62]	PVBus	Master	This is a special "id-routed" port for transmitting ATC invalidates upstream into the PCIe EndPoints, it is not a normal bus. See the parameter output_id_routed_transform. The FastModels ATC Invalidate and PRI Response protocol specifies how to route and deal with this port.
qtw_pvbus_m	PVBus	Master	Manager port used for Table Walks, MSIs and Queue access. Note that although this is a manager port, it can still receive DVM messages.
sec_override	Signal	Slave	Allow certain registers to be accessible to non-secure accesses from reset, as described in the TCU_SCR register. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sup_btm	Signal	Slave	System supports BTM and will be reflected in the IDR registers. This signal can override the value set by the parameters configuring the IDR registers. If BTM (Broadcast Table Maintenance) is not supported then DVM messages will be ignored. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sup_cohacc	Signal	Slave	System supports COHACC and will be reflected in the IDR registers. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sup_httu	Signal	Slave	System supports HTTU and will be reflected in the value of SMMU_IDR0.HTTU: - 0b00 (HTTU not supported) if sup_httu is 0 - 0b10 (update of AF and Dirty flags supported) if sup_httu is 1 The default value for this pin is 1. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sup_sev	Signal	Slave	System supports SEV and will be reflected in the IDR registers. This signal can override the value set by the parameters configuring the IDR registers. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
tbm_pvbus_m[62]	PVBus	Master	The TBU manager ports that carry transactions that have been translated from the correspondingly numbered tbs_pvbus_s[] port.
tbs_pvbus_s[62]	PVBus	Slave	The TBU subordinate ports that receive transactions to be translated. They will exit the SMMU through the same numbered pvbus_m[] port.
tbu_cri_irpt[62]	Signal	Master	Critical error interrupt for RAS events from the TBU. This is used for specific uncorrected errors where a System Coprocessor (SCP) might have to reset the system because the stability of the system can no longer be guaranteed. NOTE in the hardware then this is called ras_cri.
tbu_eri_irpt[62]	Signal	Master	Error Recovery Interrupt for RAS events from the TBU. This is used for uncorrected errors. NOTE in the hardware then this is called ras_eri.

Name	Protocol	Type	Description
tbu_fhi_irpt[62]	Signal	Master	Fault Handling Interrupt for RAS events from the TBU. Usually this is for corrected errors. NOTE in the hardware then this is called ras_fhi.
tbu_pmu_irpt[62]	Signal	Master	TBU Performance Monitoring Unit interrupt, one per TBU.
tbu_pmusnapshot_ack[62]	Signal	Master	PMU snapshot interface for the TBU, ack a snapshot.
tbu_pmusnapshot_req[62]	Signal	Slave	PMU snapshot interface for the TBU, request a snapshot.
tbu_reset_in[62]	Signal	Slave	Reset signals The TBUs can have independent reset signals. Each signal tbu_reset_in[n] corresponds to the TBU using tbs_pvbus_s[n]/tbs_pvbus_s[n] pair. If the SMMU receives a transaction whilst the TBU is expected to be in reset then it will complain using the ArchMsg.Warning.warning trace source. Those tbu_reset_in that correspond to a PCIe-RC connection can be connected to monitor the PCIe-RC's reset signal. If it receives an ATS request when in reset then it will complain in a similar way. You must connect these pins if you wish the TCU_NODE_STATUS for the nodes to be accurate (including any connected to the PCIe-RC).
tcu_cri_irpt	Signal	Master	Critical error interrupt for RAS events from the TCU. This is used for specific uncorrected errors where a System Coprocessor (SCP) might have to reset the system because the stability of the system can no longer be guaranteed. NOTE in the hardware then this is called ras_cri.
tcu_eri_irpt	Signal	Master	Error Recovery Interrupt for RAS events from the TCU. This is used for uncorrected errors. NOTE in the hardware then this is called ras_eri.
tcu_fhi_irpt	Signal	Master	Fault Handling Interrupt for RAS events from the TCU. Usually this is for corrected errors. NOTE in the hardware then this is called ras_fhi.
tcu_pmu_irpt	Signal	Master	TCU Performance Monitoring Unit interrupt
tcu_pmusnapshot_ack	Signal	Master	PMU snapshot interface for the TCU, ack a snapshot.
tcu_pmusnapshot_req	Signal	Slave	PMU snapshot interface This is a four-phase handshake to have the corresponding PMCG perform a capture of the current counter values. PMU snapshot interface for the TCU, request a snapshot.
tcu_reset_in	Signal	Slave	The reset signal to the TCU interface.

Parameters for MMU_700

TCUCFG_PARTID_WIDTH

The width of the MPAM PARTID on the bus.

See also parameter mpam_attribute_transform. Accepted Values: 1 6 9

Type

int

Default value

0x9

TCUCFG_XLATE_SLOTS

Maximum number of outstanding stalled transactions that the SMMU supports. Note that TCUCFG_XLATE_SLOTS must be \geq TCUCFG_PTW_SLOTS which is currently fixed to 512. Accepted Values: 512 1024 2048 4096

Type

int

Default value

0x200

all_error_messages_through_trace

Some conditions in the SMMU are so strange that the software programming the SMMU has done something wrong. At this point messages are output to either ArchMsg.Error.* or ArchMsg.Warning.* or to the error stream of the simulator. Outputting to the error stream of the simulator may cause it to return with a non-zero exit status.

If you set this option to true then instead of using the error stream of the simulator it will always use a trace stream allowing the simulation to exit with a zero exit status.

Type

bool

Default value

0x0

axi_stream_msi_TDEST

ID of the AXI stream subordinate port on the GIC that receives SMMU originated MSIs sent directly. The name of the GIC port is axi_stream_msi_s.

NOTE that if axi_stream_msi_addr_to_match[1:0] \neq 0, or the address is above OAS then this is effectively disabled.

See also: * axi_stream_msi_addr_to_match * axi_stream_msi_TID

Type

int

Default value

0x0

axi_stream_msi_TID

ID of the AXI stream manager port that sends SMMU TCU originated MSIs directly to the GIC. The name of the SMMU port is axi_stream_msi_m.

NOTE that if axi_stream_msi_addr_to_match[1:0] \neq 0, or the address is above OAS then this is effectively disabled.

See also: * axi_stream_msi_addr_to_match * axi_stream_msi_TDEST

Type

int

Default value

0x0

axi_stream_msi_addr_to_match

If the last two bits are 0, any SMMU-originated MSI which exactly matches the address will be sent through the AXI stream port axi_stream_msi_m which is usually connected to the GIC.

This parameter drives the value of the axi_stream_msi_addr_to_match_s port at simulation reset. For every reset after that, the value of the port will be sampled and used if changed.

NOTE that the entire address must match, including bits [1:0]. As MSIs are 32 bit aligned then if axi_stream_msi_addr_to_match[1:0] != 0, or the address is above OAS then this is effectively disabled.

See also: * axi_stream_msi_TID * axi_stream_msi_TDEST

Type

int

Default value

0xffffffffffffffff

behaviour_of_sampled_at_reset_signals

Some configuration signals into the SMMU are sampled on negedge of reset.

However, it can sometimes be hard to arrange to drive a configuration pin before the negedge of reset.

The configuration pins are sampled: 0 -- at negedge reset. 1 -- at negedge reset, but if a later change occurs at the same simulated time, and no transactions have occurred, then they will be resampled and the SMMU reset again.

Type

int

Default value

0x0

cmdq_max_number_of_commands_to_buffer

The command queues can buffer fetched commands before issuing them. This parameter is roughly the maximum number of commands to do this for. The programmer visible effects are that just because the CONS pointer shows a command has been _consumed_ does not necessarily mean that it has been issued (and completed). Higher values will accentuate this effect.

Type

int

Default value

0xa

enable_device_id_checks

If this parameter is true then the DeviceIDs seen by the GIC are: * for client devices:
 DeviceID = StreamID + translated_device_id_base * for SMMU-generated MSIs:
 smmu_msi_device_id

This parameter enables two checks:

* If the DeviceID is used in the output_attribute_transform then if it overflows 32 bits then the model will warn. If the DeviceID is not used then it is assumed that the external agent that forms the DeviceID will warn if it overflows.

* If the SMMU supports MSIs, then the model will check that the GIC will be able to distinguish an MSI generated by the SMMU from one generated by a client device.

As the exact mechanism to determine the DeviceID is in the system and not necessarily under control of the SMMU then you can disable these warnings using this parameter.

See also the parameters: output_attribute_transform and msi_attribute_transform.

Type

bool

Default value

0x1

howto_identify

If 'use-identify' then will use the 'identify' port to determine the SSD, StreamID, SubStreamID. Otherwise, this string extracts them from the transaction's attributes.

Examples:-

SEC_SID=ExtendedID[63], SSV=ExtendedID[62], SubstreamID=ExtendedID[51:32],
 StreamID=ExtendedID[31:0] nSEC_SID=ExtendedID[63], StreamID=ExtendedID[55:24],
 nSSV=ExtendedID[20], SubstreamID=ExtendedID[19:0] StreamID[31:24]=0,
 StreamID[23:0]=ExtendedID[23:0], SSV=1[0], ...

The StreamID (32 b) is valid if SIDV is 1 or both SIDV and nSIDV are unused.

SEC_SID is 1b in the model. For RME systems, in HW then SEC_SID is 2b, in the model SEC_SID_bit_1 represents bit[1].

The SubstreamID (20 b) is valid if SSV is true.

SIDV == 0 indicates a NoStreamID transaction and the SSD is the PAS of the transaction, SEC_SID is not used.

nSEC_SID, nSSV, nSIDV are available with negative logic. Different attributes are independent and can use negative or positive logic.

Type

string

Default value

SEC_SID=ExtendedID[63], SSV=ExtendedID[62], SubstreamID=ExtendedID[51:32],
StreamID=ExtendedID[31:0]

ish_is_osh_DANGER

When set, any transaction that would use the architectural inner shareable domain is converted to use the outer shareable domain.

NOTE that this parameter should match the equivalent ish_is_osh from the PE. If an incompatible value of the ish_is_osh parameter is configured for the PE and the SMMU, data coherency may be compromised.

NOTE that all implementations should have this parameter as TRUE but it is allowed in the model to give it a false value for modelling and debug purposes only.

Type

bool

Default value

0x1

list_of_ns_sid_high_at_bitpos0

A comma-separated list of values to bitwise OR into each Non-secure StreamID for each TBU/Node. Bit 0 of the value corresponds to bit 0 of the StreamID.

Each TBU that is connected to a PCIe-RC (see list_of_pcie_rc) must serve a unique contiguous subset of StreamIDs as determined by their top bits. This is used in order to know which port to route ATC Invalidates and PRI Responses to the PCIe subsystems.

The empty string corresponds to all 0s.

Type

string

Default value**list_of_pcie_mode**

A comma-separated list of ranges of ports that represent TBUs that are attached to PCIe Root Complexes (PCIe-RC). A single PCIe-RC might use several TBUs and stripe accesses across them. The attributes handling for these TBUs are slightly different in that if the PCIe transaction is NoSnoop and the output attributes of the translation would be weaker than iNC-oNC-osh then the output is forced to iNC-oNC-osh.

iNC-oNC-osh == "inner normal non-cacheable, out normal non-cacheable, outer shared"

Type

string

Default value**list_of_pcie_rc**

This is a list of ports that are connected to PCIe Root Complex (PCIe-RC) by a protocol called DTI-ATS. This port is used to transport ATS and PRI Requests to the SMMU from the PCIe-RC.

In the real hardware, then the PCIe-RC uses this port for ATS/PRI, and then the actual transactions go through separate TBUs. In the model, then this port can accept actual transactions as well. However, in the model, then the ATC Invalidates and the PRI Responses need to be transferred over the corresponding pvbush_id_routed_m port as DTI-ATS is bidirectional, but PVBush is not.

Type

string

Default value**list_of_s_sid_high_at_bitpos0**

A comma-separated list of values to bitwise OR into each Secure StreamID for each TBU/Node. Bit 0 of the value corresponds to bit 0 of the StreamID.

The empty string corresponds to all 0s.

"use-ns" can be used to apply the list_of_ns_sid_high_at_bitpos0 values instead.

Type

string

Default value**mpam_attribute_transform**

If MPAM is supported, this is applied to _all_ downstream transactions to transport the MPAM information. * ""/"none" -- no transform * How to alter the output attributes.
Example: "ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID, ExtendedID[38]=MPAM_NS"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags.

RHS Symbols: * MPAM_PARTID * MPAM_PMG * MPAM_NS * numeric literals

Any bits with no transform are unchanged.

NOTE: * attribute transforms applied before this: * for client transactions 'output_attribute_transform'. * for table walks 'tw_qs_attribute_transform'. * for MSIs 'msi_attribute_transform'. * for translated transactions from client devices then MPAM_NS = ! SEC_SID.

Type

string

Default value

ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID,
ExtendedID[38]=MPAM_NS

msi_attribute_transform

Transform downstream attributes of MSI transactions. * ""/"none" -- no transform * How to alter output attributes of SMMU-generated MSIs. Example: "UserFlags[15:0]=smmu_msi_device_id[31:16], MasterID[15:0]=smmu_msi_device_id[15:0], ExtendedID=0"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags

RHS Symbols: * the parameter smmu_msi_device_id * 'interrupt_kind' * 0/1 -- EVENTQ s/ns * 2 -- PRIQ * 3/4 -- CMD_SYNC s/ns * 5/6 -- GERROR s/ns * 7/8 -- PMCG s/ns * 9/10/11 -- RAS FHI/ERI/CRI * 12/13 -- gpf_far/gpt_cfg_far * 14/15/16/17 -- Realm EVENTQ/PRIQ/CMDQ/GERROR * HWATTR_KIND_0: PBHA information * numeric literals.

ExtendedID/MasterID/UserFlags start with values {0, 0xFFFFFFFF, 0} respectively.

Any bits with no transform are unchanged.

This transform can be used to determine the DeviceID passed to the GIC to distinguish MSIs generated by the SMMU from those generated by client devices.

NOTE: * see also 'output_attribute_transform' and enable_device_id_checks.

Type

string

Default value

ExtendedID[31:0]=smmu_msi_device_id, MasterID=0xFFFFFFFF

normalize_input_normal_non_iWB_oWB_to_iNC_oNC_osh_DANGER

When set, use Inner Non Cacheable, Outer Non Cacheable, Outer Shareable for any upstream transaction that would use any of the following attributes: - Normal Non-cacheable Bufferable - Normal Non-cacheable Non-bufferable - Write-through

NOTE that this parameter should match the equivalent configuration from the PE. If an incompatible value of this parameter is configured for the PE and the SMMU, data coherency may be compromised.

NOTE that all implementations should have this parameter as TRUE but it is allowed in the model to give it a false value for modelling and debug purposes only.

Type

bool

Default value

0x1

number_of_ports

The number of port pairs that the SMMU has.

Type

int

Default value

0x1

output_attribute_transform

Transform the downstream attributes of a translated transaction.

* ""/"none": no transform * How to alter output attributes. Example:

"ExtendedID[15:0]=DeviceID[15:0], UserFlags[31]=nSSV, UserFlags[19:0]=SubstreamID, MasterID[10]=MasterID[11], MasterID[11]=MasterID[10]"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags: incoming/outgoing attributes.

RHS Symbols: * DeviceID: StreamID + translated_device_id_base * StreamID/SubstreamID/SSV/SEC_SID * nSEC_SID/nSSV: negative logic versions. * St1PBHA/St2PBHA: Page Based Hardware Attributes from leaf descriptors (zero if unused). * STE_IMPDEF1: STE[127:116] * numeric literals.

The StreamID has had ns_sid_high/s_sid_high ORred into it for the appropriate TBU.

NOTE: * 'mpam_attribute_transform' is applied after this.

Type

string

Default value

ExtendedID[31:0]=DeviceID

output_id_routed_transform

The SMMU generates the following ID-routed transaction on the pybus_id_routed_m bus: * ATC Invalidate * PRI Response

This parameter expresses how the SMMU should express: * the StreamID * the Trusted (T) bit

The value is a comma-separated list of assignments:

Address[27:12]=StreamID[15:0], ExtendedID[60]=T, ExtendedID[15:0]=StreamID[31:16]

Address bits[11:0] cannot be used.

The LHS can be one of: * PAS * MasterID/ExtendedID/UserFlags * Address

The RHS can be one of: * a numeric constant * SSD * T or negative version nT * StreamID

For realm (or 'Trusted') transactions, then SSD=0b11, T=1, nT=0. For non-secure (or 'Non-Trusted') transactions, then SSD=0b01, T=0, nT=1

Type

string

Default value

Address[27:12]=StreamID[15:0], PAS=SSD

prefetch_only_requests

The simulator supports 'prefetch-only' DMI requests, which can occur for anytime for any reason and are intended to be invisible to the end execution of the model and to the user.

0 -- deny all prefetch-only requests
 1 -- use debug requests for any page table walks -- form and use debug TLB/cache entries -- any faults will not record, but deny the prefetch request
 2 -- treat prefetch-only requests like normal transactions -- use normal page table walk transactions -- use and form normal TLB/cache entries -- faults will alter the programmer visible state of the SMMU

0 is the safest.

1 treats the access like a debug request and requires that debug page table walks are treated correctly downstream. Any descriptors that need HTTU to allow the transaction to proceed will fail the request.

2 is dangerous, it use real transactions and reports faults that are unphysical. Real transactions can be wait()ed and this disobeys the SystemC spec for get_direct_mem_ptr().

Type

int

Default value

0x0

sec_override

The IMP DEF port sec_override controls whether some of the registers are accessible to secure or non-secure transactions. This parameter is the default value assumed for that port if the port is not driven by a signal.

Type

bool

Default value

0x0

seed

Used to seed the pseudo-random number generator that the SMMU model uses.

Type

int

Default value

0x12345678

size_of_cd_cache

The number of entries in the cache holding CD structures. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_llcd_cache

The number of entries in the cache holding L1CD descriptors. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_llste_cache

The number of entries in the cache holding L1STE descriptors. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_ste_cache

The number of entries in the cache holding STE structures. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_tlb

The number of entries in the TLB. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

smmu_msi_device_id

When appropriately enabled, assume that MSIs that are generated by the SMMU are presented to the GIC with this DeviceID.

See parameter msi_attribute_transform and enable_device_id_checks.

Type

int

Default value

0x0

sup_btm

The default value of the register field SMMU_IDR0.BTM and SMMU_ROOT_IDR0.BGPTM (when supported). This enables Broadcast TLB maintenance.

Type

bool

Default value

0x1

sup_cohacc

The default value of the register SMMU_IDR0.COHAACC

Type

bool

Default value

0x1

sup_httu

The initial value of the sup_httu port see the port description for sup_httu.

Type

bool

Default value

0x1

sup_oas

The hardware has an input port `sup_oas[2:0]` that indicates what output address size (OAS) the `_system_` has. This is sampled at reset.

The model does not have this port as it is expected to be a constant for the system and not to change. Instead it is just a parameter.

The allowed values are: 0 -- 32 bits 1 -- 36 bits 2 -- 40 bits 3 -- 42 bits 4 -- 44 bits 5 -- 48 bits 6 -- 52 bits

Type

int

Default value

0x6

sup_sev

The default value of the register `SMMU_IDR0.SEV`

Type

bool

Default value

0x1

tlb_when_do_f_tlb_conflict_on_overlap

If a TLB entry is created by a walk and it overlaps an existing entry. Then there are some architectural situations where the result is known. For all others, then an implementation is allowed to use an UNPREDICTABLE combination of the two entries, or it can generate `F_TLB_CONFLICT`:

0 -- never generate 1 -- sometimes generate 2 -- always generate

Conflicts between global and non-global entries are not detected by the model.

Type

int

Default value

0x0

translated_device_id_base

When appropriately enabled, assume that client device accesses are translated to a `DeviceID` as seen by the GIC of: `StreamID + translated_device_id_base`

See parameter `output_attribute_transform` and `enable_device_id_checks`.

Type

int

Default value

0x0

tw_qs_attribute_transform

Transform downstream attributes of table walk and queue transactions.

* ""/"none" -- no transform * How to alter the output attributes. Example:

"ExtendedID[35:32]=HWATTR_KIND_0"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags

RHS Symbols: * HWATTR_KIND_0: PBHA information * kind * for a read: * 0/1 -- L1STE/STE * 2/3 -- L1CD/CD * 4/5 -- S1/S2 TTD (including CAS) * 6 -- CMDQ * 7 -- VMS * 11/12 -- LOGPT/L1GPT * 13/14 -- LODPT/L1DPT * for a write * 0 -- EVENTQ * 1 -- PRIQ * numeric literals.

ExtendedID/MasterID/UserFlags start with values {0, 0xFFFFFFFF, 0} respectively.

Any bits with no transform are unchanged.

NOTE: * see also 'output_attribute_transform' and 'msi_attribute_transform'.

Type

string

Default value**version**

The version of this product.

Valid values are: r0p0 r1p0

Type

string

Default value

r0p0

wait_cmdq_ticks

This is the time to wait before doing something on the command queue. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFFFFF))).

Type

int

Default value

0x0

wait_eventq_ticks

This is the time to wait before doing something on the event queue. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_misc_async_actions_ticks

This is the time to wait before doing an async action that could be delayed is performed. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_msi_ticks

This is the time to wait before sending an MSI. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_pri_req_ticks

This is the time to wait before processing a PRI Request. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_pri_resp_ticks

This is the time to wait before sending a PRI Response back to the PCIe subsystem. When a PRI Response is an auto-response then the ATC might immediately make a new ATS request, that immediately fails, that immediately makes a PRI Request, that auto-responds, etc. To break this loop, then we introduce a minimum time on all PRI Responses to give other components in the system a chance to run. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x1

3.10.56 MMU_S3

This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1056: IP revisions support

Revision	Quality level
r0p0	Full support
r1p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Table 3-1057: Model quality changes

From	To
rel	r0p0=rel r1p0=pre

Table 3-1058: IP revision changes

From	To
r0p0	r0p0 r1p0

Ports added:

- ns_gbpa_abort_init
- s_gbpa_abort_init

Parameters added:

- TCUCFG_DPT_SUPPORT
- ns_gbpa_abort_init
- s_gbpa_abort_init
- size_of_dpttlb
- sup_btm
- sup_httu

- sup_sev

Iris and MTI instances for MMU_S3

This model has the following Iris instances:

Table 3-1059: MMU_S3 Iris instances

InstanceName	ComponentName
MMU_S3	MMU_S3
MMU_S3.register_file[0]	PVBusSlave
MMU_S3.service_request_tbu[0]	PVBusSlave
MMU_S3.service_request_tbu[10]	PVBusSlave
MMU_S3.service_request_tbu[11]	PVBusSlave
MMU_S3.service_request_tbu[12]	PVBusSlave
MMU_S3.service_request_tbu[13]	PVBusSlave
MMU_S3.service_request_tbu[14]	PVBusSlave
MMU_S3.service_request_tbu[15]	PVBusSlave
MMU_S3.service_request_tbu[16]	PVBusSlave
MMU_S3.service_request_tbu[17]	PVBusSlave
MMU_S3.service_request_tbu[18]	PVBusSlave
MMU_S3.service_request_tbu[19]	PVBusSlave
MMU_S3.service_request_tbu[1]	PVBusSlave
MMU_S3.service_request_tbu[20]	PVBusSlave
MMU_S3.service_request_tbu[21]	PVBusSlave
MMU_S3.service_request_tbu[22]	PVBusSlave
MMU_S3.service_request_tbu[23]	PVBusSlave
MMU_S3.service_request_tbu[24]	PVBusSlave
MMU_S3.service_request_tbu[25]	PVBusSlave
MMU_S3.service_request_tbu[26]	PVBusSlave
MMU_S3.service_request_tbu[27]	PVBusSlave
MMU_S3.service_request_tbu[28]	PVBusSlave
MMU_S3.service_request_tbu[29]	PVBusSlave
MMU_S3.service_request_tbu[2]	PVBusSlave
MMU_S3.service_request_tbu[30]	PVBusSlave
MMU_S3.service_request_tbu[31]	PVBusSlave
MMU_S3.service_request_tbu[32]	PVBusSlave
MMU_S3.service_request_tbu[33]	PVBusSlave
MMU_S3.service_request_tbu[34]	PVBusSlave
MMU_S3.service_request_tbu[35]	PVBusSlave
MMU_S3.service_request_tbu[36]	PVBusSlave
MMU_S3.service_request_tbu[37]	PVBusSlave
MMU_S3.service_request_tbu[38]	PVBusSlave

InstanceName	ComponentName
MMU_S3.service_request_tbu[39]	PVBusSlave
MMU_S3.service_request_tbu[3]	PVBusSlave
MMU_S3.service_request_tbu[40]	PVBusSlave
MMU_S3.service_request_tbu[41]	PVBusSlave
MMU_S3.service_request_tbu[42]	PVBusSlave
MMU_S3.service_request_tbu[43]	PVBusSlave
MMU_S3.service_request_tbu[44]	PVBusSlave
MMU_S3.service_request_tbu[45]	PVBusSlave
MMU_S3.service_request_tbu[46]	PVBusSlave
MMU_S3.service_request_tbu[47]	PVBusSlave
MMU_S3.service_request_tbu[48]	PVBusSlave
MMU_S3.service_request_tbu[49]	PVBusSlave
MMU_S3.service_request_tbu[4]	PVBusSlave
MMU_S3.service_request_tbu[50]	PVBusSlave
MMU_S3.service_request_tbu[51]	PVBusSlave
MMU_S3.service_request_tbu[52]	PVBusSlave
MMU_S3.service_request_tbu[53]	PVBusSlave
MMU_S3.service_request_tbu[54]	PVBusSlave
MMU_S3.service_request_tbu[55]	PVBusSlave
MMU_S3.service_request_tbu[56]	PVBusSlave
MMU_S3.service_request_tbu[57]	PVBusSlave
MMU_S3.service_request_tbu[58]	PVBusSlave
MMU_S3.service_request_tbu[59]	PVBusSlave
MMU_S3.service_request_tbu[5]	PVBusSlave
MMU_S3.service_request_tbu[60]	PVBusSlave
MMU_S3.service_request_tbu[61]	PVBusSlave
MMU_S3.service_request_tbu[62]	PVBusSlave
MMU_S3.service_request_tbu[63]	PVBusSlave
MMU_S3.service_request_tbu[6]	PVBusSlave
MMU_S3.service_request_tbu[7]	PVBusSlave
MMU_S3.service_request_tbu[8]	PVBusSlave
MMU_S3.service_request_tbu[9]	PVBusSlave
MMU_S3.tbu[0]	PVBusMapper
MMU_S3.tbu[10]	PVBusMapper
MMU_S3.tbu[11]	PVBusMapper
MMU_S3.tbu[12]	PVBusMapper
MMU_S3.tbu[13]	PVBusMapper
MMU_S3.tbu[14]	PVBusMapper
MMU_S3.tbu[15]	PVBusMapper

InstanceName	ComponentName
MMU_S3.tbu[16]	PVBusMapper
MMU_S3.tbu[17]	PVBusMapper
MMU_S3.tbu[18]	PVBusMapper
MMU_S3.tbu[19]	PVBusMapper
MMU_S3.tbu[1]	PVBusMapper
MMU_S3.tbu[20]	PVBusMapper
MMU_S3.tbu[21]	PVBusMapper
MMU_S3.tbu[22]	PVBusMapper
MMU_S3.tbu[23]	PVBusMapper
MMU_S3.tbu[24]	PVBusMapper
MMU_S3.tbu[25]	PVBusMapper
MMU_S3.tbu[26]	PVBusMapper
MMU_S3.tbu[27]	PVBusMapper
MMU_S3.tbu[28]	PVBusMapper
MMU_S3.tbu[29]	PVBusMapper
MMU_S3.tbu[2]	PVBusMapper
MMU_S3.tbu[30]	PVBusMapper
MMU_S3.tbu[31]	PVBusMapper
MMU_S3.tbu[32]	PVBusMapper
MMU_S3.tbu[33]	PVBusMapper
MMU_S3.tbu[34]	PVBusMapper
MMU_S3.tbu[35]	PVBusMapper
MMU_S3.tbu[36]	PVBusMapper
MMU_S3.tbu[37]	PVBusMapper
MMU_S3.tbu[38]	PVBusMapper
MMU_S3.tbu[39]	PVBusMapper
MMU_S3.tbu[3]	PVBusMapper
MMU_S3.tbu[40]	PVBusMapper
MMU_S3.tbu[41]	PVBusMapper
MMU_S3.tbu[42]	PVBusMapper
MMU_S3.tbu[43]	PVBusMapper
MMU_S3.tbu[44]	PVBusMapper
MMU_S3.tbu[45]	PVBusMapper
MMU_S3.tbu[46]	PVBusMapper
MMU_S3.tbu[47]	PVBusMapper
MMU_S3.tbu[48]	PVBusMapper
MMU_S3.tbu[49]	PVBusMapper
MMU_S3.tbu[4]	PVBusMapper
MMU_S3.tbu[50]	PVBusMapper

InstanceName	ComponentName
MMU_S3.tb[51]	PVBusMapper
MMU_S3.tb[52]	PVBusMapper
MMU_S3.tb[53]	PVBusMapper
MMU_S3.tb[54]	PVBusMapper
MMU_S3.tb[55]	PVBusMapper
MMU_S3.tb[56]	PVBusMapper
MMU_S3.tb[57]	PVBusMapper
MMU_S3.tb[58]	PVBusMapper
MMU_S3.tb[59]	PVBusMapper
MMU_S3.tb[5]	PVBusMapper
MMU_S3.tb[60]	PVBusMapper
MMU_S3.tb[61]	PVBusMapper
MMU_S3.tb[62]	PVBusMapper
MMU_S3.tb[63]	PVBusMapper
MMU_S3.tb[6]	PVBusMapper
MMU_S3.tb[7]	PVBusMapper
MMU_S3.tb[8]	PVBusMapper
MMU_S3.tb[9]	PVBusMapper

This model has the following MTI trace components:

Table 3-1060: MMU_S3 MTI instances

InstanceName	ComponentName
MMU_S3	MMU_S3
MMU_S3.register_file[0]	PVBusSlave
MMU_S3.service_request_tbu[0]	PVBusSlave
MMU_S3.service_request_tbu[10]	PVBusSlave
MMU_S3.service_request_tbu[11]	PVBusSlave
MMU_S3.service_request_tbu[12]	PVBusSlave
MMU_S3.service_request_tbu[13]	PVBusSlave
MMU_S3.service_request_tbu[14]	PVBusSlave
MMU_S3.service_request_tbu[15]	PVBusSlave
MMU_S3.service_request_tbu[16]	PVBusSlave
MMU_S3.service_request_tbu[17]	PVBusSlave
MMU_S3.service_request_tbu[18]	PVBusSlave
MMU_S3.service_request_tbu[19]	PVBusSlave
MMU_S3.service_request_tbu[1]	PVBusSlave
MMU_S3.service_request_tbu[20]	PVBusSlave
MMU_S3.service_request_tbu[21]	PVBusSlave
MMU_S3.service_request_tbu[22]	PVBusSlave

InstanceName	ComponentName
MMU_S3.service_request_tbu[23]	PVBusSlave
MMU_S3.service_request_tbu[24]	PVBusSlave
MMU_S3.service_request_tbu[25]	PVBusSlave
MMU_S3.service_request_tbu[26]	PVBusSlave
MMU_S3.service_request_tbu[27]	PVBusSlave
MMU_S3.service_request_tbu[28]	PVBusSlave
MMU_S3.service_request_tbu[29]	PVBusSlave
MMU_S3.service_request_tbu[2]	PVBusSlave
MMU_S3.service_request_tbu[30]	PVBusSlave
MMU_S3.service_request_tbu[31]	PVBusSlave
MMU_S3.service_request_tbu[32]	PVBusSlave
MMU_S3.service_request_tbu[33]	PVBusSlave
MMU_S3.service_request_tbu[34]	PVBusSlave
MMU_S3.service_request_tbu[35]	PVBusSlave
MMU_S3.service_request_tbu[36]	PVBusSlave
MMU_S3.service_request_tbu[37]	PVBusSlave
MMU_S3.service_request_tbu[38]	PVBusSlave
MMU_S3.service_request_tbu[39]	PVBusSlave
MMU_S3.service_request_tbu[3]	PVBusSlave
MMU_S3.service_request_tbu[40]	PVBusSlave
MMU_S3.service_request_tbu[41]	PVBusSlave
MMU_S3.service_request_tbu[42]	PVBusSlave
MMU_S3.service_request_tbu[43]	PVBusSlave
MMU_S3.service_request_tbu[44]	PVBusSlave
MMU_S3.service_request_tbu[45]	PVBusSlave
MMU_S3.service_request_tbu[46]	PVBusSlave
MMU_S3.service_request_tbu[47]	PVBusSlave
MMU_S3.service_request_tbu[48]	PVBusSlave
MMU_S3.service_request_tbu[49]	PVBusSlave
MMU_S3.service_request_tbu[4]	PVBusSlave
MMU_S3.service_request_tbu[50]	PVBusSlave
MMU_S3.service_request_tbu[51]	PVBusSlave
MMU_S3.service_request_tbu[52]	PVBusSlave
MMU_S3.service_request_tbu[53]	PVBusSlave
MMU_S3.service_request_tbu[54]	PVBusSlave
MMU_S3.service_request_tbu[55]	PVBusSlave
MMU_S3.service_request_tbu[56]	PVBusSlave
MMU_S3.service_request_tbu[57]	PVBusSlave
MMU_S3.service_request_tbu[58]	PVBusSlave

InstanceName	ComponentName
MMU_S3.service_request_tbu[59]	PVBusSlave
MMU_S3.service_request_tbu[5]	PVBusSlave
MMU_S3.service_request_tbu[60]	PVBusSlave
MMU_S3.service_request_tbu[61]	PVBusSlave
MMU_S3.service_request_tbu[62]	PVBusSlave
MMU_S3.service_request_tbu[63]	PVBusSlave
MMU_S3.service_request_tbu[6]	PVBusSlave
MMU_S3.service_request_tbu[7]	PVBusSlave
MMU_S3.service_request_tbu[8]	PVBusSlave
MMU_S3.service_request_tbu[9]	PVBusSlave
MMU_S3.tbu[0]	PVBusMapper
MMU_S3.tbu[10]	PVBusMapper
MMU_S3.tbu[11]	PVBusMapper
MMU_S3.tbu[12]	PVBusMapper
MMU_S3.tbu[13]	PVBusMapper
MMU_S3.tbu[14]	PVBusMapper
MMU_S3.tbu[15]	PVBusMapper
MMU_S3.tbu[16]	PVBusMapper
MMU_S3.tbu[17]	PVBusMapper
MMU_S3.tbu[18]	PVBusMapper
MMU_S3.tbu[19]	PVBusMapper
MMU_S3.tbu[1]	PVBusMapper
MMU_S3.tbu[20]	PVBusMapper
MMU_S3.tbu[21]	PVBusMapper
MMU_S3.tbu[22]	PVBusMapper
MMU_S3.tbu[23]	PVBusMapper
MMU_S3.tbu[24]	PVBusMapper
MMU_S3.tbu[25]	PVBusMapper
MMU_S3.tbu[26]	PVBusMapper
MMU_S3.tbu[27]	PVBusMapper
MMU_S3.tbu[28]	PVBusMapper
MMU_S3.tbu[29]	PVBusMapper
MMU_S3.tbu[2]	PVBusMapper
MMU_S3.tbu[30]	PVBusMapper
MMU_S3.tbu[31]	PVBusMapper
MMU_S3.tbu[32]	PVBusMapper
MMU_S3.tbu[33]	PVBusMapper
MMU_S3.tbu[34]	PVBusMapper
MMU_S3.tbu[35]	PVBusMapper

InstanceName	ComponentName
MMU_S3.tb[36]	PVBusMapper
MMU_S3.tb[37]	PVBusMapper
MMU_S3.tb[38]	PVBusMapper
MMU_S3.tb[39]	PVBusMapper
MMU_S3.tb[3]	PVBusMapper
MMU_S3.tb[40]	PVBusMapper
MMU_S3.tb[41]	PVBusMapper
MMU_S3.tb[42]	PVBusMapper
MMU_S3.tb[43]	PVBusMapper
MMU_S3.tb[44]	PVBusMapper
MMU_S3.tb[45]	PVBusMapper
MMU_S3.tb[46]	PVBusMapper
MMU_S3.tb[47]	PVBusMapper
MMU_S3.tb[48]	PVBusMapper
MMU_S3.tb[49]	PVBusMapper
MMU_S3.tb[4]	PVBusMapper
MMU_S3.tb[50]	PVBusMapper
MMU_S3.tb[51]	PVBusMapper
MMU_S3.tb[52]	PVBusMapper
MMU_S3.tb[53]	PVBusMapper
MMU_S3.tb[54]	PVBusMapper
MMU_S3.tb[55]	PVBusMapper
MMU_S3.tb[56]	PVBusMapper
MMU_S3.tb[57]	PVBusMapper
MMU_S3.tb[58]	PVBusMapper
MMU_S3.tb[59]	PVBusMapper
MMU_S3.tb[5]	PVBusMapper
MMU_S3.tb[60]	PVBusMapper
MMU_S3.tb[61]	PVBusMapper
MMU_S3.tb[62]	PVBusMapper
MMU_S3.tb[63]	PVBusMapper
MMU_S3.tb[6]	PVBusMapper
MMU_S3.tb[7]	PVBusMapper
MMU_S3.tb[8]	PVBusMapper
MMU_S3.tb[9]	PVBusMapper

MMU_S3 contains the following CADI targets:

- MMU_S3

About MMU_S3

Architectural features and microarchitectural registers match those of the corresponding revision:

r0p0

Aligns with r0p0-00eac0

r1p0

Aligns with r1p0-00dev0

For a list of model limitations and usage notes, see `$PVLIB_HOME/LISA/SMMU_S3.lisa`.

Security State Determination (SSD)

The model uses the term SSD to mean the security state that the transaction, register, structure, or event belongs to. In the SMMUv3 architecture, the sideband signal SEC_SID holds this information for the transactions, but uses a different encoding.

Before RME, SEC_SID was a one-bit signal. With RME, in the hardware, it was expanded to two bits. To retain backwards compatibility in the model, SEC_SID remains as one-bit in the parameter `howto_identify`, but the second bit can be expressed with `SEC_SID_bit_1`, and its negative logic version `nSEC_SID_bit_1`.

Table 3-1061: Security State Determination (SSD) encoding

Security state	SSD	SEC_SID_bit_1	SEC_SID
Secure	0	0	1
Non-secure	1	0	0
Root (reserved)	2	1	1
Realm	3	1	0

Ports for MMU_S3

Table 3-1062: Ports

Name	Protocol	Type	Description
<code>axi_stream_msi_addr_to_match_s</code>	<code>Value_64</code>	Slave	Any SMMU-originated MSI which exactly matches the address in this port will be sent through the AXI stream port <code>axi_stream_msi_m</code> which is usually connected to the GIC through <code>axi_stream_msi_s</code> . As MSIs are 32 bit aligned then if <code>bits[1:0] != 0</code> or the address is above OAS then this is effectively disabled. NOTE that the model does not support <code>tcu_sid[31:0]</code> which is the MSI DeviceID to send on <code>axi_stream_msi_m</code> . Instead the parameter <code>smmu_msi_device_id</code> is used. See also the parameters: <code>axi_stream_msi_TID</code> and <code>axi_stream_msi_TDEST</code> . The default value of this port is set by the parameter <code>axi_stream_msi_addr_to_match</code> . This port is sampled at <code>negedge</code> of <code>tcu_reset_in</code> and must be set before the <code>negedge</code> of the reset signal.
<code>axi_stream_msi_m</code>	<code>PVBus</code>	Master	Manager port used for sending SMMU originated MSIs directly to the GIC
<code>clk_in</code>	<code>ClockSignal</code>	Slave	Clock signal (in RTL <code>aclk</code>) This is a clock time-base used by the TCU to spread some of its processing over time, if enabled by the <code>wait_*</code> parameters. The clock must always be connected.

Name	Protocol	Type	Description
cmd_sync_irpt_ns	Signal	Master	Pulsed interrupt output signal for non-secure CMD_SYNC having a completion signal of SIG_IRQ.
cmd_sync_irpt_r	Signal	Master	Pulsed interrupt output signal for realm CMD_SYNC having a completion signal of SIG_IRQ. This pin exists in r0 but is tied off to 0. It is implemented in r1.
cmd_sync_irpt_s	Signal	Master	Pulsed interrupt output signal for secure CMD_SYNC having a completion signal of SIG_IRQ.
event_q_irpt_ns	Signal	Master	Pulsed interrupt output signal for the non-secure event queue becoming non-empty.
event_q_irpt_r	Signal	Master	Pulsed interrupt output signal for the realm event queue becoming non-empty. This pin exists in r0 but is tied off to 0. It is implemented in r1.
event_q_irpt_s	Signal	Master	Pulsed interrupt output signal for the secure event queue becoming non-empty.
evento	Signal	Master	Event signal This aligns with the eventoreq signal on the RTL. The eventoack signal is not supported.
global_irpt_ns	Signal	Master	Pulsed interrupt output signal for non-secure SMMU_GERROR(N) signalling an error.
global_irpt_r	Signal	Master	Pulsed interrupt output signal for realm SMMU_R_GERROR(N) signalling an error. This pin exists in r0 but is tied off to 0. It is implemented in r1.
global_irpt_s	Signal	Master	Pulsed interrupt output signal for secure SMMU_S_GERROR(N) signalling an error.
gpf_far	Signal	Master	An error becomes active in SMMU_ROOT_GPF_FAR.
gpt_cfg_far	Signal	Master	An error becomes active in SMMU_ROOT_GPT_CFG_FAR.
identify	SMMUv3AEMIdentifyProtocol	Master	Map the transaction to the tuple (StreamID, SubStreamID, SubStreamIDValid, SSD) The StreamID that is produced by the implementation of this protocol is not the final StreamID. The final StreamID is produced by using the list_of_ns_sid_high_at_bitpos0/list_of_s_sid_high_at_bitpos0 parameter to map the StreamID based on the upstream port index. Also see the parameter howto_identify which can replace the functionality of this port under certain circumstances.
logptsz_s	Value	Slave	RME: This is a four bit signal that encodes the region size that a single LOGPT entry covers. The default value of this port is derived from the parameter rme_logpt_entry_covers_log2size_in_bytes which is in a different format to the port. If a valid value is driven then it will be put in the field SMMU_ROOT_GPT_BASE_CFG.LOGPTSZ. The port uses the same encoding as the field. If an invalid value is driven to this port and legacy_tz_en is low then the value is reported in the SMMU_ROOT_GPT_BASE_CFG.LOGPTSZ and then all transactions will fault with a GPT Configuration fault (gpt_cfg_far). This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.

Name	Protocol	Type	Description
legacy_tz_en	Signal	Slave	Tie this high to get non-RME behaviour. On the real hardware, then each of the TCUs and the TBUs have a legacy_tz_en and they must all be driven to the same value. In the model, we only have a single version of this pin. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
ns_gbpa_abort_init	Signal	Slave	This port is an Non-secure global bypass. The ns_gbpa_abort_init signal sets the reset value of SMMU_GBPA.ABORT: 0 - On reset, do not abort all incoming transactions. 1 - On reset, abort all incoming transactions. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
pri_q_irpt_ns	Signal	Master	Pulsed interrupt output signal for the non-secure PRI queue becoming non-empty.
pri_q_irpt_r	Signal	Master	Pulsed interrupt output signal for the realm PRI queue becoming non-empty. This pin exists in r0 but is tied off to 0. It is implemented in r1.
prog_pvbus_s	PVBus	Slave	Register subordinate port (in RTL PROG)
pvbus_id_routed_m[62]	PVBus	Master	This is a special "id-routed" port for transmitting ATC invalidates upstream into the PCIe EndPoints, it is not a normal bus. See parameter output_id_routed_transform. The FastModels ATC Invalidate and PRI Response protocol specifies how to route and deal with this port.
qtw_pvbus_m	PVBus	Master	Manager port used for Table Walks, MSIs and Queue access. Note that although this is a manager port, it can still receive DVM messages.
s_gbpa_abort_init	Signal	Slave	This port is an secure global bypass. The s_gbpa_abort_init signal sets the reset value of SMMU_S_GBPA.ABORT: 0 - On reset, do not abort all incoming transactions. 1 - On reset, abort all incoming transactions. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sec_override	Signal	Slave	Allow certain registers to be accessible to non-secure accesses from reset, as described in the TCU_SCR register. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sup_btm	Signal	Slave	System supports BTM and will be reflected in the IDR registers. This signal can override the value set by the parameters configuring the IDR registers. If BTM (Broadcast Table Maintenance) is not supported then DVM messages will be ignored. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sup_cohacc	Signal	Slave	System supports COHACC and will be reflected in the IDR registers. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
sup_httu	Signal	Slave	System supports HTTU and will be reflected in the value of SMMU_IDR0.HTTU: - 0b00 (HTTU not supported) if sup_httu is 0 - 0b10 (update of AF and Dirty flags supported) if sup_httu is 1 The default value for this pin is 1. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.

Name	Protocol	Type	Description
sup_sev	Signal	Slave	System supports SEV and will be reflected in the IDR registers. This signal can override the value set by the parameters configuring the IDR registers. This port is sampled at negedge of tcu_reset_in and must be set before the negedge of the reset signal.
tbm_pvbus_m[62]	PVBus	Master	The TBU manager ports that carry transactions that have been translated from the correspondingly numbered tbs_pvbus_s[] port.
tbs_pvbus_s[62]	PVBus	Slave	The TBU subordinate ports that receive transactions to be translated. They will exit the SMMU through the same numbered pvbus_m[] port.
tbu_crit_err[62]	Signal	Master	Critical error. This cannot occur in the model except by using an Integration Register to generate it.
tbu_pmu_irpt[62]	Signal	Master	TBU Performance Monitoring Unit interrupt, one per TBU.
tbu_pmusnapshot_ack[62]	Signal	Master	PMU snapshot interface for the TBU, ack a snapshot.
tbu_pmusnapshot_req[62]	Signal	Slave	PMU snapshot interface for the TBU, request a snapshot.
tbu_ras_cri[62]	Signal	Master	Critical error interrupt for RAS events from the TBU. This is used for specific uncorrected errors where a System Coprocessor (SCP) might have to reset the system because the stability of the system can no longer be guaranteed. NOTE that in the MMU-700 model this is called tbu_cri_irpt.
tbu_ras_eri[62]	Signal	Master	Error Recovery Interrupt for RAS events from the TBU. This is used for uncorrected errors. NOTE that in the MMU-700 model this is called tbu_eri_irpt. NOTE that in the MMU-700 model this is called tbu_eri_irpt.
tbu_ras_fhi[62]	Signal	Master	Fault Handling Interrupt for RAS events from the TBU. Usually this is for corrected errors. NOTE that in the MMU-700 model this is called tbu_fhi_irpt.
tbu_ras_lt_cri[62]	Signal	Master	Level triggered critical error interrupt for RAS events from the TBU.
tbu_ras_lt_eri[62]	Signal	Master	Level triggered error recovery interrupt for RAS events from the TBU.
tbu_ras_lt_fhi[62]	Signal	Master	Level triggered fault handling interrupt for RAS events from the TBU.
tbu_ras_lt_irpt_v[62]	Signal	Master	Level triggered valid output for connection to System RAS agents. Asserted when the RAS record contains at least one valid error.
tbu_reset_in[62]	Signal	Slave	Reset signals The TBUs can have independent reset signals. Each signal tbu_reset_in[n] corresponds to the TBU using tbs_pvbus_s[n]/tbs_pvbus_s[n] pair. If the SMMU receives a transaction whilst the TBU is expected to be in reset then it will complain using the ArchMsg.Warning.warning trace source. Those tbu_reset_in that correspond to a PCIe-RC connection can be connected to monitor the PCIe-RC's reset signal. If it receives an ATS request when in reset then it will complain in a similar way. You must connect these pins if you wish the TCU_NODE_STATUS for the nodes to be accurate (including any connected to the PCIe-RC).

Name	Protocol	Type	Description
tcu_pmu_irpt	Signal	Master	TCU Performance Monitoring Unit interrupt
tcu_pmusnapshot_ack	Signal	Master	PMU snapshot interface for the TCU, ack a snapshot.
tcu_pmusnapshot_req	Signal	Slave	PMU snapshot interface This is a four-phase handshake to have the corresponding PMCG perform a capture of the current counter values. PMU snapshot interface for the TCU, request a snapshot.
tcu_ras_cri	Signal	Master	Critical error interrupt for RAS events from the TCU. This is used for specific uncorrected errors where a System Coprocessor (SCP) might have to reset the system because the stability of the system can no longer be guaranteed. NOTE that in the MMU-700 model this is called tcu_cri_irpt.
tcu_ras_eri	Signal	Master	Error Recovery Interrupt for RAS events from the TCU. This is used for uncorrected errors. NOTE that in the MMU-700 model this is called tcu_eri_irpt.
tcu_ras_fhi	Signal	Master	Fault Handling Interrupt for RAS events from the TCU. Usually this is for corrected errors. NOTE that in the MMU-700 model this is called tcu_fhi_irpt.
tcu_ras_lt_cri	Signal	Master	Level triggered critical error interrupt for RAS events from the TCU.
tcu_ras_lt_eri	Signal	Master	Level triggered error recovery interrupt for RAS events from the TCU.
tcu_ras_lt_fhi	Signal	Master	Level triggered fault handling interrupt for RAS events from the TCU.
tcu_ras_lt_irpt_v	Signal	Master	Level triggered valid output for connection to System RAS agents. Asserted when the RAS record contains at least one valid error.
tcu_reset_in	Signal	Slave	The reset signal to the TCU interface.

Parameters for MMU_S3

TCUCFG_DPT_SUPPORT

MMU-S3 r1 Parameter Only: Enable Device Permission Table (DPT), a mechanism to enforce the association between granules of physical address space and the memory footprint of virtual machines.

0 -- Realm DPT is disabled 1 -- Realm DPT is enabled (default)

Type

bool

Default value

0x1

TCUCFG_DVM_VAS

Virtual address size used by the system. Once set, this value is discoverable via TCU_SYSDISC35.TCUCFG_DVM_VAS.

In HW, it is important to get this parameter correct as it determines the DVM message format. If this doesn't match the PEs, DVM messages will be misinterpreted and any TLBI operations performed will be incorrectly applied.

The model uses a representation of DVM that does not depend on the VA size and so misconfiguring this has no effect other than on the system discovery register value. Accepted Values: 49 53

Type

int

Default value

0x35

TCUCFG_MECID_WIDTH

Memory Encryption Context (MEC) is a new feature introduced in MMU-S3. The MECID is a 1-16 bit identifier that, if implemented, supports Memory Encryption Contexts for the Realm programming interface. The given value indicates the number of bits in the MECID. A value of 0 will disable MEC. Accepted Values: 0 4 8 12 16

Type

int

Default value

0x10

TCUCFG_PARTID_WIDTH

The width of the MPAM PARTID on the bus.

The value 10 is just for MMU_S3 r1.

See also parameter mpam_attribute_transform. Accepted Values: 1 6 9 10

Type

int

Default value

0x9

TCUCFG_XLATE_SLOTS

Maximum number of outstanding stalled transactions that the SMMU supports Note that TCUCFG_XLATE_SLOTS must be \geq TCUCFG_PTW_SLOTS which currently fixed to 512. Accepted Values: 512 1024 2048 4096

Type

int

Default value

0x200

all_error_messages_through_trace

Some conditions in the SMMU are so strange that the software programming the SMMU has done something wrong. At this point messages are output to either ArchMsg.Error.* or ArchMsg.Warning.* or to the error stream of the simulator. Outputting to the error stream of the simulator may cause it to return with a non-zero exit status.

If you set this option to true then instead of using the error stream of the simulator it will always use a trace stream allowing the simulation to exit with a zero exit status.

Type

bool

Default value

0x0

axi_stream_msi_TDEST

ID of the AXI stream subordinate port on the GIC that receives SMMU originated MSIs sent directly. The name of the GIC port is axi_stream_msi_s.

NOTE that if axi_stream_msi_addr_to_match[1:0] != 0, or the address is above OAS then this is effectively disabled.

See also: * axi_stream_msi_addr_to_match * axi_stream_msi_TID

Type

int

Default value

0x0

axi_stream_msi_TID

ID of the AXI stream manager port that sends SMMU TCU originated MSIs directly to the GIC. The name of the SMMU port is axi_stream_msi_m.

NOTE that if axi_stream_msi_addr_to_match[1:0] != 0, or the address is above OAS then this is effectively disabled.

See also: * axi_stream_msi_addr_to_match * axi_stream_msi_TDEST

Type

int

Default value

0x0

axi_stream_msi_addr_to_match

If the last two bits are 0, any SMMU-originated MSI which exactly matches the address will be sent through the AXI stream port axi_stream_msi_m which is usually connected to the GIC.

This parameter drives the value of the `axi_stream_msi_addr_to_match_s` port at simulation reset. For every reset after that, the value of the port will be sampled and used if changed.

NOTE that the entire address must match, including bits [1:0]. As MSIs are 32 bit aligned then if `axi_stream_msi_addr_to_match[1:0] != 0`, or the address is above OAS then this is effectively disabled.

See also: * `axi_stream_msi_TID` * `axi_stream_msi_TDEST`

Type

int

Default value

0xffffffffffffffff

behaviour_of_sampled_at_reset_signals

Some configuration signals into the SMMU are sampled on negedge of reset.

However, it can sometimes be hard to arrange to drive a configuration pin before the negedge of reset.

The configuration pins are sampled: 0 -- at negedge reset. 1 -- at negedge reset, but if a later change occurs at the same simulated time, and no transactions have occurred, then they will be resampled and the SMMU reset again.

Type

int

Default value

0x0

cmdq_max_number_of_commands_to_buffer

The command queues can buffer fetched commands before issuing them. This parameter is roughly the maximum number of commands to do this for. The programmer visible effects are that just because the CONS pointer shows a command has been `_consumed_` does not necessarily mean that it has been issued (and completed). Higher values will accentuate this effect.

Type

int

Default value

0xa

enable_device_id_checks

If this parameter is true then the DeviceIDs seen by the GIC are: * for client devices: `DeviceID = StreamID + translated_device_id_base` * for SMMU-generated MSIs: `smmu_msi_device_id`

This parameter enables two checks:

* If the DeviceID is used in the output_attribute_transform then if it overflows 32 bits then the model will warn. If the DeviceID is not used then it is assumed that the external agent that forms the DeviceID will warn if it overflows.

* If the SMMU supports MSIs, then the model will check that the GIC will be able to distinguish an MSI generated by the SMMU from one generated by a client device.

As the exact mechanism to determine the DeviceID is in the system and not necessarily under control of the SMMU then you can disable these warnings using this parameter.

See also the parameters: output_attribute_transform and msi_attribute_transform.

Type

bool

Default value

0x1

hide_warning_EOPD_differs_from_what_would_be_cached

When this parameter is set to true, warnings that the effective EOPD value differs from what would be cached in the TLB are disabled. False (warnings are showed) by default.

Type

bool

Default value

0x0

hide_warning_NoStreamID_transaction_for_unsupported_PAS_or_MPAM_SP

When RME is not supported then a NoStreamID transaction with PAS[1] == 1 or MPAM_SP[1] == 1 is treated as though PAS[1] == 0 and MPAM_SP[1] == 0. This is usually a system construction error and is not expected to occur.

The SMMU will warn when this occurs, but the warning can be hidden by setting this parameter.

Type

bool

Default value

0x0

howto_identify

If 'use-identify' then will use the 'identify' port to determine the SSD, StreamID, SubStreamID. Otherwise, this string extracts them from the transaction's attributes.

Examples:-

SEC_SID=ExtendedID[63], SSV=ExtendedID[62], SubstreamID=ExtendedID[51:32],
StreamID=ExtendedID[31:0] nSEC_SID=ExtendedID[63], StreamID=ExtendedID[55:24],

nSSV=ExtendedID[20], SubstreamID=ExtendedID[19:0] StreamID[31:24]=0,
StreamID[23:0]=ExtendedID[23:0], SSV=1[0], ...

The StreamID (32 b) is valid if SIDV is 1 or both SIDV and nSIDV are unused.

SEC_SID is 1b in the model. For RME systems, in HW then SEC_SID is 2b, in the model SEC_SID_bit_1 represents bit[1].

The SubstreamID (20 b) is valid if SSV is true.

SIDV == 0 indicates a NoStreamID transaction and the SSD is the PAS of the transaction, SEC_SID is not used.

nSEC_SID, nSSV, nSIDV are available with negative logic. Different attributes are independent and can use negative or positive logic.

Type

string

Default value

SEC_SID=ExtendedID[63], SSV=ExtendedID[62], SubstreamID=ExtendedID[51:32],
StreamID=ExtendedID[31:0]

howto_identify_NoStreamID_extra_info

The behavior of this parameter depends on 'howto_identify' * if it equals 'use-identify' then this must be "", otherwise there is an error. * if it identifies a NoStreamID transaction (SIDV=0) then this parameter includes one or more of * MPAM_SP * MPAM_PARTID * MPAM_PMG * MECID * HWATTR_KIND_0 * in any other case, this parameter is ignored. Fields set in this parameter MUST NOT overlap the SIDV/nSIDV fields in 'howto_identify'

Example:-

MPAM_PMG[7:0]=ExtendedID[62:55], MPAM_PARTID[15:0]=ExtendedID[54:39],
MPAM_SP[1:0]=ExtendedID[38:37], MECID[15:0]=UserFlags[31:16]
HWATTR_KIND_0[3:0]=ExtendedID[42:39]

Type

string

Default value

ish_is_osh_DANGER

When set, any transaction that would use the architectural inner shareable domain is converted to use the outer shareable domain.

NOTE that this parameter should match the equivalent ish_is_osh from the PE. If an incompatible value of the ish_is_osh parameter is configured for the PE and the SMMU, data coherency may be compromised.

NOTE that all implementations should have this parameter as TRUE but it is allowed in the model to give it a false value for modelling and debug purposes only.

Type

bool

Default value

0x1

legacy_tz_en

The default value of the legacy_tz_en pin:

0 -- RME is enabled 1 -- RME is disabled

Type

bool

Default value

0x0

list_of_ns_sid_high_at_bitpos0

A comma-separated list of values to bitwise OR into each Non-secure StreamID for each TBU/Node. Bit 0 of the value corresponds to bit 0 of the StreamID.

Each TBU that is connected to a PCIe-RC (see list_of_pcie_rc) must serve a unique contiguous subset of StreamIDs as determined by their top bits. This is used in order to know which port to route ATC Invalidates and PRI Responses to the PCIe subsystems. The effective value for any PCIe-RC ports must be the same for non-secure and realm. See list_of_r_sid_high_at_bitpos0.

The empty string corresponds to all 0s.

Type

string

Default value**list_of_pcie_mode**

A comma-separated list of ranges of ports that represent TBUs that are attached to PCIe Root Complexes (PCIe-RC). A single PCIe-RC might use several TBUs and stripe accesses across them. The attributes handling for these TBUs are slightly different in that if the PCIe transaction is NoSnoop and the output attributes of the translation would be weaker than iNC-oNC-osh then the output is forced to iNC-oNC-osh.

iNC-oNC-osh == "inner normal non-cacheable, out normal non-cacheable, outer shared"

Type

string

Default value

list_of_pcie_rc

This is a list of ports that are connected to PCIe Root Complex (PCIe-RC) by a protocol called DTI-ATS. This port is used to transport ATS and PRI Requests to the SMMU from the PCIe-RC.

In the real hardware, then the PCIe-RC uses this port for ATS/PRI, and then the actual transactions go through separate TBUs. In the model, then this port can accept actual transactions as well. However, in the model, then the ATC Invalidates and the PRI Responses need to be transferred over the corresponding pvbush_id_routed_m port as DTI-ATS is bidirectional, but PVBush is not.

Type

string

Default value**list_of_r_sid_high_at_bitpos0**

A comma-separated list of values to bitwise OR into each Realm StreamID for each TBU/Node. Bit 0 of the value corresponds to bit 0 of the StreamID. This only has an effect for r1 and later.

Each TBU that is connected to a PCIe-RC (see list_of_pcie_rc) must serve a unique contiguous subset of StreamIDs as determined by their top bits. This is used in order to know which port to route ATC Invalidates and PRI Responses to the PCIe subsystems. The effective value for any PCIe-RC ports must be the same for non-secure and realm. See list_of_ns_sid_high_at_bitpos0.

The empty string corresponds to all 0s.

"use-ns" can be used to apply the list_of_ns_sid_high_at_bitpos0 values instead.

Type

string

Default value**list_of_s_sid_high_at_bitpos0**

A comma-separated list of values to bitwise OR into each Secure StreamID for each TBU/Node. Bit 0 of the value corresponds to bit 0 of the StreamID.

The empty string corresponds to all 0s.

"use-ns" can be used to apply the list_of_ns_sid_high_at_bitpos0 values instead.

Type

string

Default value

mec_attribute_transform

If MEC is supported, this is applied to `_all_` downstream transactions to transport the MEC information. * `""/"none"` -- no transform * How to alter the output attributes. Example: `"UserFlags[31:16]=MECID[15:0]"`

RHS/LHS Symbols: * `ExtendedID/MasterID/UserFlags`.

RHS Symbols: * `MECID` * numeric literals.

Any bits with no transform are unchanged.

NOTE: * attribute transforms applied before this: * for client transactions `'output_attribute_transform'/'output_attribute_transform_for_NoStreamID'`. * for table walks `'tw_qs_attribute_transform'`. * for MSIs `'msi_attribute_transform'`. * if MPAM is enabled `'mpam_attribute_transform'`.

Type

string

Default value**mpam_attribute_transform**

If MPAM is supported, this is applied to `_all_` downstream transactions to transport the MPAM information. * `""/"none"` -- no transform * How to alter the output attributes. Example: `"ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID, ExtendedID[38]=MPAM_SP[0]"`

RHS/LHS Symbols: * `ExtendedID/MasterID/UserFlags`.

RHS Symbols: * `MPAM_PARTID` * `MPAM_PMG` * `MPAM_NS` * `MPAM_SP` * numeric literals

Any bits with no transform are unchanged.

NOTE: * attribute transforms applied before this: * for client transactions `'output_attribute_transform' / 'output_attribute_transform_for_NoStreamID'`. * for table walks `'tw_qs_attribute_transform'`. * for MSIs `'msi_attribute_transform'`. * `'mec_attribute_transform'` is applied after this. * for translated transactions from client devices then `MPAM_NS = ! SEC_SID`.

Type

string

Default value

`ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID, ExtendedID[38]=MPAM_NS`

msi_attribute_transform

Transform downstream attributes of MSI transactions. * `""/"none"` -- no transform * How to alter output attributes of SMMU-generated MSIs. Example:

"UserFlags[15:0]=smmu_msi_device_id[31:16], MasterID[15:0]=smmu_msi_device_id[15:0], ExtendedID=0"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags

RHS Symbols: * the parameter smmu_msi_device_id * 'interrupt_kind' * 0/1 -- EVENTQ s/ns * 2 -- PRIQ * 3/4 -- CMD_SYNC s/ns * 5/6 -- GERROR s/ns * 7/8 -- PMCG s/ns * 9/10/11 -- RAS FHI/ERI/CRI * 12/13 -- gpf_far/gpt_cfg_far * 14/15/16/17 -- Realm EVENTQ/PRIQ/CMDQ/GERROR * HWATTR_KIND_0: PBHA information * numeric literals.

ExtendedID/MasterID/UserFlags start with values {0, 0xFFFFFFFF, 0} respectively.

Any bits with no transform are unchanged.

This transform can be used to determine the DeviceID passed to the GIC to distinguish MSIs generated by the SMMU from those generated by client devices.

NOTE: * see also 'output_attribute_transform' and enable_device_id_checks.

Type

string

Default value

ExtendedID[31:0]=smmu_msi_device_id, MasterID=0xFFFFFFFF

normalize_input_normal_non_iWB_oWB_to_iNC_oNC_osh_DANGER

When set, use Inner Non Cacheable, Outer Non Cacheable, Outer Shareable for any upstream transaction that would use any of the following attributes: - Normal Non-cacheable Bufferable - Normal Non-cacheable Non-bufferable - Write-through

NOTE that this parameter should match the equivalent configuration from the PE. If an incompatible value of this parameter is configured for the PE and the SMMU, data coherency may be compromised.

NOTE that all implementations should have this parameter as TRUE but it is allowed in the model to give it a false value for modelling and debug purposes only.

Type

bool

Default value

0x1

ns_gbpa_abort_init

The default value of the tie off signal 'ns_gbpa_abort_init'

Type

bool

Default value

0x0

number_of_ports

The number of port pairs that the SMMU has.

Type

int

Default value

0x1

output_attribute_transform

Transform the downstream attributes of a translated transaction.

* ""/"none": no transform * How to alter output attributes. Example:

"ExtendedID[15:0]=DeviceID[15:0], UserFlags[31]=nSSV, UserFlags[19:0]=SubstreamID, MasterID[10]=MasterID[11], MasterID[11]=MasterID[10]"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags: incoming/outgoing attributes.

RHS Symbols: * DeviceID: StreamID + translated_device_id_base * StreamID/SubstreamID/SSV/SEC_SID * nSEC_SID/nSSV: negative logic versions. * St1PBHA/St2PBHA: Page Based Hardware Attributes from leaf descriptors (zero if unused). * STE_IMPDEF1: STE[127:116] * numeric literals.

The StreamID has had ns_sid_high/s_sid_high ORred into it for the appropriate TBU.

NOTE: * 'mpam_attribute_transform' is applied after this.

Type

string

Default value

ExtendedID[31:0]=DeviceID

output_attribute_transform_for_NoStreamID

Transform downstream attributes of NoStreamID transactions.

* ""/"none": no transform * How to alter output attributes. Example:

"ExtendedID[15:0]=0, UserFlags[31]=1, UserFlags[19:0]=0, MasterID[10]=MasterID[11], MasterID[11]=MasterID[10] MasterID[9:6]=HWATTR_KIND_0"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags: incoming/outgoing attributes.

RHS Symbols: * SIDV = 0, nSIDV = 1 (fixed values to indicate NoStreamID) * PAS * HWATTR_KIND_0 * numeric literals.

Any bits with no transform are unchanged.

NOTE: * 'mpam_attribute_transform' and 'mec_attribute_transform' are applied in order after this. * see also 'output_attribute_transform' for StreamID transactions.

Type

string

Default value

ExtendedID[31:0]=0, ExtendedID[32]=1

output_id_routed_transform

The SMMU generates the following ID-routed transaction on the pvbus_id_routed_m bus: * ATC Invalidate * PRI Response

This parameter expresses how the SMMU should express: * the StreamID * the Trusted (T) bit

The value is a comma-separated list of assignments:

Address[27:12]=StreamID[15:0], ExtendedID[60]=T, ExtendedID[15:0]=StreamID[31:16]

Address bits[11:0] cannot be used.

The LHS can be one of: * PAS * MasterID/ExtendedID/UserFlags * Address

The RHS can be one of: * a numeric constant * SSD * T or negative version nT * StreamID

For realm (or 'Trusted') transactions, then SSD=0b11, T=1, nT=0. For non-secure (or 'Non-Trusted') transactions, then SSD=0b01, T=0, nT=1

Type

string

Default value

Address[27:12]=StreamID[15:0], PAS=SSD

prefetch_only_requests

The simulator supports 'prefetch-only' DMI requests, which can occur for anytime for any reason and are intended to be invisible to the end execution of the model and to the user.

0 -- deny all prefetch-only requests 1 -- use debug requests for any page table walks -- form and use debug TLB/cache entries -- any faults will not record, but deny the prefetch request 2 -- treat prefetch-only requests like normal transactions -- use normal page table walk transactions -- use and form normal TLB/cache entries -- faults will alter the programmer visible state of the SMMU

0 is the safest.

1 treats the access like a debug request and requires that debug page table walks are treated correctly downstream. Any descriptors that need HTTU to allow the transaction to proceed will fail the request.

2 is dangerous, it use real transactions and reports faults that are unphysical. Real transactions can be wait()ed and this disobeys the SystemC spec for get_direct_mem_ptr().

Type

int

Default value

0x0

rme_l0gpt_entry_covers_log2size_in_bytes

Each LOGPT entry covers: $2^{**rme_l0gpt_entry_covers_log2size_in_bytes}$ bytes of address space.

The valid values for this parameter are: * 30, 34, 36, 39

This parameter is reported in an encoded format as the read-only field: SMMU_ROOT_GPT_BASE_CFG.LOGPTSZ

This parameter can be overridden by the port l0gptsz_s when sampled on negedge of reset.

Type

int

Default value

0x1e

s_gbpa_abort_init

The default value of the tie off signal 's_gbpa_abort_init'

Type

bool

Default value

0x0

sec_override

The IMP DEF port sec_override controls whether some of the registers are accessible to secure or non-secure/realm transactions. This parameter is the default value assumed for that port if the port is not driven by a signal.

Type

bool

Default value

0x0

seed

Used to seed the pseudo-random number generator that the SMMU model uses.

Type

int

Default value

0x12345678

size_of_cd_cache

The number of entries in the cache holding CD structures. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_dpttlb

The number of entries in the DPT TLB. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_llcd_cache

The number of entries in the cache holding L1CD descriptors. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_llste_cache

The number of entries in the cache holding L1STE descriptors. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_ste_cache

The number of entries in the cache holding STE structures. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_tlb

The number of entries in the TLB. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

smmu_msi_device_id

When appropriately enabled, assume that MSIs that are generated by the SMMU are presented to the GIC with this DeviceID.

See parameter msi_attribute_transform and enable_device_id_checks.

Type

int

Default value

0x0

sup_btm

The default value of the register field SMMU_IDR0.BTM and SMMU_ROOT_IDR0.BGPTM (when supported). This enables Broadcast TLB maintenance.

Type

bool

Default value

0x1

sup_cohacc

The default value of the register SMMU_IDR0.COHAACC

Type

bool

Default value

0x1

sup_httu

The initial value of the sup_httu port see the port description for sup_httu.

Type

bool

Default value

0x1

sup_oas

The hardware has an input port sup_oas[2:0] that indicates what output address size (OAS) the _system_ has. This is sampled at reset.

The model does not have this port as it is expected to be a constant for the system and not to change. Instead it is just a parameter.

The allowed values are: 0 -- 32 bits 1 -- 36 bits 2 -- 40 bits 3 -- 42 bits 4 -- 44 bits 5 -- 48 bits 6 -- 52 bits

Type

int

Default value

0x6

sup_sev

The default value of the register SMMU_IDR0.SEV

Type

bool

Default value

0x1

tlb_when_do_f_tlb_conflict_on_overlap

If a TLB entry is created by a walk and it overlaps an existing entry. Then there are some architectural situations where the result is known. For all others, then an implementation is allowed to use an UNPREDICTABLE combination of the two entries, or it can generate F_TLB_CONFLICT:

0 -- never generate 1 -- sometimes generate 2 -- always generate

Conflicts between global and non-global entries are not detected by the model.

Type

int

Default value

0x0

translated_device_id_base

When appropriately enabled, assume that client device accesses are translated to a DeviceID as seen by the GIC of: StreamID + translated_device_id_base

See parameter output_attribute_transform and enable_device_id_checks.

Type

int

Default value

0x0

tw_qs_attribute_transform

Transform downstream attributes of table walk and queue transactions.

* ""/"none" -- no transform * How to alter the output attributes. Example:

"ExtendedID[35:32]=HWATTR_KIND_0"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags

RHS Symbols: * HWATTR_KIND_0: PBHA information * kind * for a read: * 0/1 -- L1STE/STE * 2/3 -- L1CD/CD * 4/5 -- S1/S2 TTD (including CAS) * 6 -- CMDQ * 7 -- VMS * 11/12 -- LOGPT/L1GPT * 13/14 -- LODPT/L1DPT * for a write * 0 -- EVENTQ * 1 -- PRIQ * numeric literals.

ExtendedID/MasterID/UserFlags start with values {0, 0xFFFFffff, 0} respectively.

Any bits with no transform are unchanged.

NOTE: * see also 'output_attribute_transform' and 'msi_attribute_transform'.

Type

string

Default value**version**

The version of this product.

Valid values are: r0p0 r1p0

Type

string

Default value

r0p0

wait_cmdq_ticks

This is the time to wait before doing something on the command queue. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_eventq_ticks

This is the time to wait before doing something on the event queue. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_misc_async_actions_ticks

This is the time to wait before doing an async action that could be delayed is performed. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_msi_ticks

This is the time to wait before sending an MSI. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_pri_req_ticks

This is the time to wait before processing a PRI Request. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_pri_resp_ticks

This is the time to wait before sending a PRI Response back to the PCIe subsystem. When a PRI Response is an auto-response then the ATC might immediately make a new ATS request, that immediately fails, that immediately makes a PRI Request, that auto-responds, etc. To break this loop, then we introduce a minimum time on all PRI Responses to give other

components in the system a chance to run. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x1

3.10.57 MemoryMappedCounterModule

Memory Mapped Counter Module for Generic Timers. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1063: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- `has_additional_registers`

Iris and MTI instances for MemoryMappedCounterModule

This model has the following Iris instances:

Table 3-1064: MemoryMappedCounterModule Iris instances

InstanceName	ComponentName
MemoryMappedCounterModule	MemoryMappedCounterModule
MemoryMappedCounterModule.pvbus_control_s[0]	PVBusSlave
MemoryMappedCounterModule.pvbus_read_s[0]	PVBusSlave

This model has the following MTI trace components:

Table 3-1065: MemoryMappedCounterModule MTI instances

InstanceName	ComponentName
MemoryMappedCounterModule	MemoryMappedCounterModule
MemoryMappedCounterModule.pvbus_control_s[0]	PVBusSlave
MemoryMappedCounterModule.pvbus_read_s[0]	PVBusSlave

MemoryMappedCounterModule contains the following CADI targets:

- MemoryMappedCounterModule

About MemoryMappedCounterModule

This component must be used by multicluster models. It also must be used to run a single core system with a timer that runs at a rate that is different to the input clock to the core.



The component has two bus slave ports because the architecture specification permits you to map each set of registers at different, non-contiguous base addresses.

Ports for MemoryMappedCounterModule

Table 3-1066: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	This clock input determines the frequency of the Physical Count provided to the clusters connected to the cntvalueb port.
cntvalueb	CounterInterface	Master	This master port implements a private protocol between the cluster and the MemoryMappedCounterModule. This must be connected to the cntvalueb port on each cluster in the system and to the MemoryMappedCounterModule component.
counter_reset	Signal	Slave	Resets when set.
pvbust_control_s	PVBus	Slave	This slave port provides memory-mapped read write access to the control registers of the module.
pvbust_read_s	PVBus	Slave	This slave port provides memory-mapped read access to the status frame registers.

Parameters for MemoryMappedCounterModule

base_frequency

Reset value for CNTFID0, base frequency in Hz

Type

int

Default value

0x5f5e100

cntcldr0123_C

Values to be returned for control-frame CIDR registers

Type

int

Default value

0x0

cntcldr0123_R

Values to be returned for read-frame CIDR registers

Type

int

Default value

0x0

cntpidr0123_C

Values to be returned for control-frame PIDR registers 0-3

Type

int

Default value

0x0

cntpidr0123_R

Values to be returned for read-frame PIDR registers 0-3

Type

int

Default value

0x0

cntpidr4567_C

Values to be returned for control-frame PIDR registers 4-7

Type

int

Default value

0x0

cntpidr4567_R

Values to be returned for read-frame PIDR registers 4-7

Type

int

Default value

0x0

diagnostics

Diagnostics

Type

int

Default value

0x0

has_additional_registers

Implements additional REFCLK CNT control registers

Type

bool

Default value

0x0

has_counter_scaling

Implements ARMv8.4 generic counter scaling (FEAT_CNTSC).

Type

bool

Default value

0x0

non_arch_fixed_frequency

If set, ignore CNTFID0 and instead use this frequency in Hz

Type

int

Default value

0x0

non_arch_start_at_default

Firmware is expected to enable the timer at boot time. However, turning this parameter on is a model-specific way of enabling the counter module out of reset.

Type

bool

Default value

0x0

readonly_is_WI

Ignore (rather than failing) on writes to read-frame

Type

bool

Default value

0x0

use_real_time****Deprecated, this parameter will be removed in future versions**** Update the Generic Timer counter at a real-time base frequency instead of simulator time**Type**

bool

Default value

0x0

3.10.58 MessageHandlingUnit

Message Handling Unit. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1067: IP revisions support

Revision	Quality level
v2.0	Full support
v2.1	Full support
v3	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for MessageHandlingUnit

This model has the following Iris instances:

Table 3-1068: MessageHandlingUnit Iris instances

InstanceName	ComponentName
MessageHandlingUnit	MessageHandlingUnit
MessageHandlingUnit.a_to_b_v2	MessageHandlingUnitV2
MessageHandlingUnit.a_to_b_v3	MessageHandlingUnitV3
MessageHandlingUnit.version_mapper_a_to_b_rec	PVBusMapper
MessageHandlingUnit.version_mapper_a_to_b_snd	PVBusMapper

This model has the following MTI trace components:

Table 3-1069: MessageHandlingUnit MTI instances

InstanceName	ComponentName
MessageHandlingUnit.a_to_b_v2	MessageHandlingUnitV2
MessageHandlingUnit.a_to_b_v3	MessageHandlingUnitV3
MessageHandlingUnit.version_mapper_a_to_b_rec	PVBusMapper
MessageHandlingUnit.version_mapper_a_to_b_snd	PVBusMapper

MessageHandlingUnit contains the following CADI targets:

- MessageHandlingUnit
- MessageHandlingUnitV2
- MessageHandlingUnitV3

Ports for MessageHandlingUnit

Table 3-1070: Ports

Name	Protocol	Type	Description
pvbus_s_rec	PVBus	Slave	-
pvbus_s_snd	PVBus	Slave	-
rec_combined_channel_irq_out[200]	Signal	Master	-
rec_combined_irq_out	Signal	Master	-
rec_reset_in	Signal	Slave	-
snd_combined_channel_irq_out[200]	Signal	Master	-
snd_combined_irq_out	Signal	Master	-
snd_reset_in	Signal	Slave	-

Parameters for MessageHandlingUnit

NUM_DB_CH
Number of doorbell channels
Type
int
Default value
0x1

NUM_FAST_CH
Number of fast channels
Type
int
Default value
0x1

a_to_b_v3.NUM_FIFO_CH
Number of FIFO Channels, default=1
Type
int
Default value
0x1

a_to_b_v3.auto_op_full
AutoOp mode - AutoOp(min) == false, AutoOp(full) == true - default: AutoOp(min)
Type
bool
Default value
0x0

a_to_b_v3.fast_ch_group_int_enable

Fast Channel group interrupts enable, default=false

Type

bool

Default value

0x0

a_to_b_v3.fifo_depth

Depth of the FIFO = fifo_depth + 1

Type

int

Default value

0x4

a_to_b_v3.m16ba_spt

Mailbox 16 bit access support to FIFO registers

Type

bool

Default value

0x0

a_to_b_v3.m32ba_spt

Mailbox 32 bit access support to FIFO registers

Type

bool

Default value

0x1

a_to_b_v3.m64ba_spt

Mailbox 64 bit access support to FIFO registers

Type

bool

Default value

0x0

a_to_b_v3.m8ba_spt

Mailbox 8 bit access support to FIFO registers

Type

bool

Default value

0x0

a_to_b_v3.monolithic

Monolithic or Distributed MHU - default: monolithic(true)

Type

bool

Default value

0x1

a_to_b_v3.p16ba_spt

Postbox 16 bit access support to FIFO registers

Type

bool

Default value

0x0

a_to_b_v3.p32ba_spt

Postbox 32 bit access support to FIFO registers

Type

bool

Default value

0x1

a_to_b_v3.p64ba_spt

Postbox 64 bit access support to FIFO registers

Type

bool

Default value

0x0

a_to_b_v3.p8ba_spt

Postbox 8 bit access support to FIFO registers

Type

bool

Default value

0x0

diagnostics

Diagnostics 0==FATAL_ERROR -> 4==DEBUG

Type

int

Default value

0x2

fast_ch_group_int_enable

Fast Channel group interrupts enable, default=false

Type

bool

Default value

0x0

fast_ch_n_per_group

Fast Channel num channels per group, default=1

Type

int

Default value

0x1

fast_ch_num_groups

Fast Channel num of groups, default=1

Type

int

Default value

0x1

fast_ch_word_size

Fast Channel word size 32bit or 64bit, default=32

Type

int

Default value

0x20

major_version

MHU major version (default=2)

Type

int

Default value

0x2

mhu_arch_beta01

true = Aligns to MHUv3.beta01; false = Aligns to MHUv3.2

Type

bool

Default value

0x0

minor_version
MHU minor version (default=1)

Type
int

Default value
0x1

product_id
MHU part number

Type
int

Default value
0x0

3.10.59 MessageHandlingUnitV2

Message Handling Unit Version 2. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1071: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for MessageHandlingUnitV2

This model has the following Iris instances:

Table 3-1072: MessageHandlingUnitV2 Iris instances

InstanceName	ComponentName
MessageHandlingUnitV2	MessageHandlingUnitV2

This model has the following MTI trace components:

Table 3-1073: MessageHandlingUnitV2 MTI instances

InstanceName	ComponentName
MessageHandlingUnitV2	MessageHandlingUnitV2

MessageHandlingUnitV2 contains the following CADI targets:

- MessageHandlingUnitV2

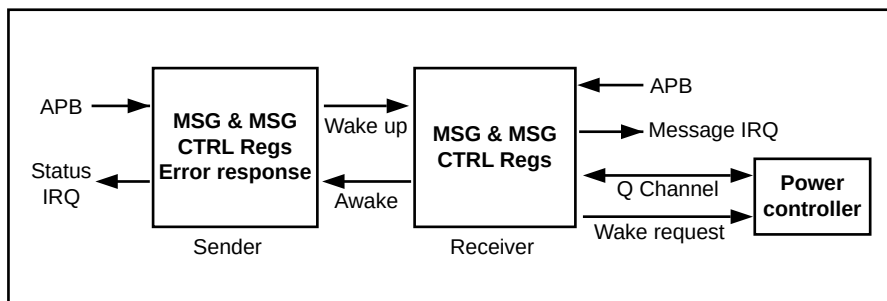
About MessageHandlingUnitV2

MessageHandlingUnitV2 is a unidirectional message channel with two APB interfaces, one for the sender and one for the receiver. There is one message handling unit for each sender and receiver pair.

The receiver can be powered off. A Q-Channel is provided for power control on the receiver side.

The message payload can be written directly to status registers. Multiple channels can be combined into a single message.

Figure 3-6: MessageHandlingUnitV2 structure



Ports for MessageHandlingUnitV2

Table 3-1074: Ports

Name	Protocol	Type	Description
int_access_nr2r	Signal	Master	-
int_access_r2nr	Signal	Master	-
mhu_combined_irq	Signal	Master	-
mhu_irq[124]	Signal	Master	-
mhu_snd_irq[124]	Signal	Master	-
pvbus_s_rec	PVBus	Slave	-
pvbus_s_snd	PVBus	Slave	-
qchannel_mhu_pwr	PChannel	Slave	-
reset_rec	Signal	Slave	-
reset_snd	Signal	Slave	-
snd_combined_irq	Signal	Master	-
wakerequest	Signal	Master	-

Parameters for MessageHandlingUnitV2

NUM_CH	Number of device channels
Type	int
Default value	0x1
minor_revision	MHUV2 minor revision, 0 for v2.0, 1 for v2.1
Type	int
Default value	0x0
product_id	MHUV2 Product ID, MHU Part Number
Type	int
Default value	0x0

3.10.60 MessageHandlingUnitV3

Message Handling Unit Version 3. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1075: IP revisions support

Revision	Quality level
v3	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for MessageHandlingUnitV3

This model has the following Iris instances:

Table 3-1076: MessageHandlingUnitV3 Iris instances

InstanceName	ComponentName
MessageHandlingUnitV3	MessageHandlingUnitV3

This model has the following MTI trace components:

Table 3-1077: MessageHandlingUnitV3 MTI instances

InstanceName	ComponentName
MessageHandlingUnitV3	MessageHandlingUnitV3

MessageHandlingUnitV3 contains the following CADI targets:

- MessageHandlingUnitV3

Ports for MessageHandlingUnitV3

Table 3-1078: Ports

Name	Protocol	Type	Description
pvbuss_rec	PVBus	Slave	Register access for Receiver/Mailbox
pvbuss_snd	PVBus	Slave	Register access for Sender/Postbox
rec_combined_irq_out	Signal	Master	All interrupts combined for Receiver/MBX
rec_fast_channel_group_irq_out[32]	Signal	Master	Receiver fast channel group interrupts
rec_fast_channel_irq_out[1024]	Signal	Master	Receiver fast channel interrupts
rec_reset_in	Signal	Slave	Reset signal for Receiver/Mailbox
snd_combined_irq_out	Signal	Master	All Interrupts combined for Sender/PBX
snd_reset_in	Signal	Slave	Reset signal for Sender/Postbox

Parameters for MessageHandlingUnitV3

NUM_DB_CH

Number of DoorBell Channels, default=1

Type

int

Default value

0x1

NUM_FAST_CH

Number of Fast Channels, default=1

Type

int

Default value

0x1

NUM_FIFO_CH

Number of FIFO Channels, default=1

Type

int

Default value

0x1

auto_op_full

AutoOp mode - AutoOp(min) == false, AutoOp(full) == true - default: AutoOp(min)

Type

bool

Default value

0x0

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG), default=2

Type

int

Default value

0x2

fast_ch_group_int_enable

Fast Channel group interrupts enable, default=false

Type

bool

Default value

0x0

fast_ch_n_per_group

Fast Channel num channels per group, default=1

Type

int

Default value

0x1

fast_ch_num_groups

Fast Channel num of groups, default=1

Type

int

Default value

0x1

fast_ch_word_size

Fast Channel word size 32bit or 64bit, default=32

Type

int

Default value

0x20

fifo_depth

Depth of the FIFO = fifo_depth + 1

Type

int

Default value

0x4

m16ba_spt

Mailbox 16 bit access support to FIFO registers

Type

bool

Default value

0x0

m32ba_spt

Mailbox 32 bit access support to FIFO registers

Type

bool

Default value

0x1

m64ba_spt

Mailbox 64 bit access support to FIFO registers

Type

bool

Default value

0x0

m8ba_spt

Mailbox 8 bit access support to FIFO registers

Type

bool

Default value

0x0

mhu_arch_beta01

true = Aligns to MHUv3.beta01; false = Aligns to MHUv3.2

Type

bool

Default value

0x0

monolithic

Monolithic or Distributed MHU - default: monolithic(true)

Type

bool

Default value

0x1

p16ba_spt

Postbox 16 bit access support to FIFO registers

Type

bool

Default value

0x0

p32ba_spt

Postbox 32 bit access support to FIFO registers

Type

bool

Default value

0x1

p64ba_spt

Postbox 64 bit access support to FIFO registers

Type

bool

Default value

0x0

p8ba_spt

Postbox 8 bit access support to FIFO registers

Type

bool

Default value

0x0

3.10.61 NI700

NI700 Interconnect Fast Model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1079: IP revisions support

Revision	Quality level
r0p0	Preliminary support
r1p0	Preliminary support
r2p0	Preliminary support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for NI700

This model has the following Iris instances:

Table 3-1080: NI700 Iris instances

InstanceName	ComponentName
NI700	NI700
NI700.decoder	PVBusMapper

This model has the following MTI trace components:

Table 3-1081: NI700 MTI instances

InstanceName	ComponentName
NI700	NI700
NI700.decoder	PVBusMapper

NI700 contains the following CADI targets:

- NI700

Configuring and using the model

- For information about the changes in this release, see the Fast Models Portfolio release notes.
- Major IP revisions (rX) are modeled and are controlled by the `revision` parameter.
- Minor IP revisions (pY) are not modeled. There are no functional differences in the model between pY revisions, with the exception of the `periph_id2` register.
- To configure the model, you must have installed Arm® Socrates™. The `mesh_config_file` parameter defines the mesh placement of NI700 components. Set it to the name of the IYML configuration file emitted by Socrates. You must use version r1p5-03rel0 or later of Socrates to generate the file. For more information about Socrates, see [Arm Socrates](#) on Arm Developer. To download it, contact [Arm Technical Support](#).
- Interconnect models are based on the TRM description and do not typically model RTL defects.
- The mapping between the port number for ASNI, AMNI, HSNI, HMNI, and PMNI ports and the xxNI name used in the yml file is its index when all xxNIs of that same type are sorted in ascending alphabetical order. For example, if the yml file has three ASNIs `asni_s100_scp`, `asni_s101_dap`, and `asni_s204_periph0`:
 - `asni_s100_scp` is mapped to `pvbuss_s_asni[0]`

- `asni_s101_dap` is mapped to `pvbuss_s_asni[1]`
- `asni_s204_periph0` is mapped to `pvbuss_s_asni[2]`

Similarly:

- AMNIs are mapped to `pvbuss_m_amni`
- HSNIs are mapped to `pvbuss_s_hsni`
- HMNIs are mapped to `pvbuss_m_hmni`
- PMNIs are mapped to `pvbuss_m_pmni`

Additionally, NI700's parser prints the name-to-index mappings when the component parameter `print_parser_log=true`.

The following functionality is expected to work:

- The discovery feature to determine the system address of all nodes.
- Hashed and non-hashed memory regions. They are parsed from the `mesh_config_file`.
- MPAM support. Software must configure MPAM override in ASNI nodes by enabling and configuring the `ASNI_AR_MPAM_OVERRIDE` (0x0E0) and `ASNI_AW_MPAM_OVERRIDE` (0x0E4) registers. The `GT_MPAM_SUPPORT` signal is ignored. Software must configure MPAM support in ASNI nodes by enabling and configuring the Request MPAM Override (0x0E0) register.
- IDM support. The IDM features Access control and Reset control are modeled. Starting in r1p0, non-secure versions of the `ACCESS_STATUS` and `RESET_STATUS` registers are present. The DeviceID and the information whether an xxNI has IDM enabled are parsed from the `mesh_config_file`. When an xxNI is isolated with IDM Access Control or under reset with IDM Reset, all transactions to and from that xxNI are aborted. With respect to IDM reset support, IDM reset signals are modeled and they should be connected to the managed devices that are connected to the respective xxNI port. The register `IDM_RESET_CONTROL` is supported. The target xxNI always enters or exits IDM reset immediately and drives the reset signals accordingly. In register `IDM_RESET_STATUS`, the bitfields `active_write` and `active_read` read always zero. In registers `IDM_RESET_READID` and `IDM_RESET_WRITEID`, the bitfields `vmaster_id` and `master_id` read always zero.
- There are no software functional differences for r2p1 and r2p0 can be used in its place.

Model limitations

- Out of scope:
 - PMU counters are not supported. Counter registers are implemented as RAZ.
 - QoS is not supported and all related registers are RAZ/WI.
 - Error injection and error generation are not supported. All error registers are RAZ/WI.
 - Power, clock, and interrupt signals are not supported.
- An access to a reserved or nonexistent register does not abort. It returns `pv::Tx_Data::TX_OK` and is RAZ/WI.
- The maximum number of manager Network Interfaces is 127. The maximum tested is 127 AMNIs, 127 PMNIs, and 9 HMNIs.

- The maximum number of subordinate Network Interfaces is 128. The maximum tested is 128 ASNIs and 9 HSNIs.
- The maximum voltage, power, and clock domains of 32 each have not been tested.
- There is no support for 1 stripe target in a group, additional granularities, or the additional stripe group remap functionality described in r2p0 TRM section 2.4.5.
- Remapping features not supported:
 - Tested 5 out of the maximum of 8 remap states.
 - The priority for multiple address remapping states.
 - One target remapped to a different target.
 - A single target remapped to a stripe group.
 - One stripe group remapped to a different stripe group.
 - A stripe group remapped to a single target.
- Stripe features not supported:
 - ASNI striping to an HMNI target has not been tested.
 - Limited testing of stripe groups with different numbers of targets and granularities.
 - Single target stripe.
- The hashed memory regions support is limited by Fast Models DMI. Due to the 4KB memory pages in DMI, granularities smaller than 4KB are not accounted for by the model. Thus, subsequent accesses within a 4KB address range are delivered to the same destination node.
- PMNI supports multiple interfaces but in the model only a single interface is listed in the `PMNI_INTERFACEID` registers of a PMNI.
- AMNI nodes in the model are interface-indifferent and registers do not reflect the protocol version.
- r2 CMO Response control is not supported.
- There is no revision string for r2p1. r2p0 is functionally equivalent.
- Hashing of stripe groups is limited to a granularity of 4096B.
- xSNI access to CFGNI is not limited by router connectivity defined in the `mesh_config_file`. It considers only whether the xSNI has the CFGNI target defined in its memory map.
- A reset after model startup does not reset the registers or address remap selections.
- `*_IDM_RESET_STRAP` and its effect on the endpoint soft reset and `IDM_RESET_CONTROL` register is not supported.
- IDM for power domains is not supported.
- No register visibility support for a debugger.
- No software control for the CMO terminate response in register `AMNI_CONFIG_CTL`.

Ports for NI700

Table 3-1082: Ports

Name	Protocol	Type	Description
idm_reset_signal_amni[127]	Signal	Master	IDM reset signals to AMNIs.
idm_reset_signal_asni[128]	Signal	Master	IDM reset signals to ASNIs.
idm_reset_signal_hmni[127]	Signal	Master	IDM reset signals to HMNIs.
idm_reset_signal_hsni[128]	Signal	Master	IDM reset signals to HSNIs.
idm_reset_signal_pmni[127]	Signal	Master	IDM reset signals to PMNIs.
pvbus_m_amni[127]	PVBus	Master	AMNI downstream ports.
pvbus_m_hmni[127]	PVBus	Master	HMNI downstream ports.
pvbus_m_pmni[127]	PVBus	Master	PMNI downstream ports.
pvbus_s_asni[128]	PVBus	Slave	ASNI upstream ports.
pvbus_s_hsni[128]	PVBus	Slave	HSNI upstream ports.
reset_in	Signal	Slave	Reset signal.

Parameters for NI700

mesh_config_file

Name of a file containing mesh placement of NI700 components.

Type

string

Default value

/tmp/plgbuild/abs_build/1378878_61223/trunk/work/fastsim/autodocs/.../RegTests/
Interconnect/NI700/ni700_premium.yml

mpam_attributes

User-defined transform to be applied to bus attributes like MasterID, ExtendedID or UserFlags, for MPAM Attributes encoded into bus attributes. For example, 'ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID, ExtendedID[38]=MPAM_NS'. An empty string disables MPAM support.

Type

string

Default value

periphbase

Value for PERIPHBASE.

Type

int

Default value

0xffffffffffffffffffff

print_config
Enables printing the config register addresses.

Type
bool

Default value
0x0

print_parser_log
Enables printing the yaml config parser log messages.

Type
bool

Default value
0x0

revision
Component revision. Currently supports r0p0, r1p0, r2p0.

Type
string

Default value
r2p0

show_banner
Show component banner: 0=>supress entire banner, 1=>suppress config file, 2+=>show full banner.

Type
int

Default value
0x2

3.10.62 PL011_Uart

ARM PrimeCell UART(PL011). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1083: IP revisions support

Revision	Quality level
r1p4	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL011_Uart

This model has the following Iris instances:

Table 3-1084: PL011_Uart Iris instances

InstanceName	ComponentName
PL011_Uart	PL011_Uart
PL011_Uart.busslave	PVBusSlave
PL011_Uart.clk_divider	ClockDivider
PL011_Uart.timer	ClockTimerThread
PL011_Uart.timer.timer	ClockTimerThread64
PL011_Uart.timer.timer.thread	SchedulerThread
PL011_Uart.timer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-1085: PL011_Uart MTI instances

InstanceName	ComponentName
PL011_Uart	PL011_Uart
PL011_Uart.busslave	PVBusSlave
PL011_Uart.clk_divider	ClockDivider

PL011_Uart contains the following CADI targets:

- ClockDivider
- ClockTimerThread
- ClockTimerThread64
- PL011_Uart
- SchedulerThread
- SchedulerThreadEvent

Using in_file and in_file_escape_sequence parameters

The UART reads input from `in_file`. If `in_file` contains a line beginning:

```
## WaitForPrompt <something-up-to-end-of-line>
```

then the UART stops reading from `in_file` until the prompt has appeared.

For example, if `in_file` contains the following lines, the UART outputs `ls` only after the root prompt appears:

```
## WaitForPrompt root #
ls
```



Use the parameter `in_file_escape_sequence` to set a different escape sequence to `##`.

Using the `untimed_fifos` parameter

When the `untimed_fifos` parameter is false, characters of serial data are clocked to or from the `SerialData` port at a rate controlled by the `clk_in_ref` clock rate and the baud-rate-divider configuration of the UART clock. Enabling `untimed_fifos` permits serial data to be sent or received as fast as it can be generated or consumed. The modem control signals are still generated correctly, so the UART is not able to transmit data faster than the receiving end can handle. For example, `TelnetTerminal` uses the CTS signal to avoid overflowing its TCP/IP buffer. See [3.7.48 TelnetTerminal](#) on page 2993.

Differences between the model and the RTL

This component does not implement the DMA functionality of the PL011 PrimeCell.

Ports for PL011_Uart

Table 3-1086: Ports

Name	Protocol	Type	Description
<code>clk_in_ref</code>	<code>ClockSignal</code>	Slave	Clock input, typically 14.745MHz, which sets the master transmit/receive rate.
<code>intr</code>	<code>Signal</code>	Master	Interrupt signal.
<code>pvbuss</code>	<code>PVBus</code>	Slave	Slave port for register access.
<code>serial_out</code>	<code>SerialData</code>	Master	Serial input/output and control signals. Used to communicate with a serial device, such as a terminal.

Parameters for PL011_Uart

`baud_rate`

Baud rate.

Type

int

Default value

0x9600

`clk_divider.div`

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clk_divider.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clock_rate

Clock rate for PL011.

Type

int

Default value

0xe10000

enable_dc4

Enable DC4 commands (try echo -e "help\024" in a Linux shell in a serial console)

Type

bool

Default value

0x1

flow_ctrl_mask_en

Enable hardware flow control workaround which forcefully disables CTSen and RTSen bits in UARTCR register

Type

bool

Default value

0x0

halt

Halt instead of shutdown for shutdown_on_eot and shutdown_tag

Type

bool

Default value

0x0

in_file

Input file for data to be read by the UART

Type

string

Default value

in_file_escape_sequence

Input file escape sequence

Type

string

Default value

##

out_file

Output file to hold data written by the UART (use '-' to send all output to stdout)

Type

string

Default value**revision**

Revision to simulate

Type

string

Default value

r1p4

shutdown_on_eot

Shutdown simulation when a EOT (ASCII 4) char is transmitted (useful for regression tests when semihosting is not available)

Type

bool

Default value

0x0

shutdown_tag

Shutdown simulation when a string is transmitted

Type

string

Default value**toggle_mti**

Start/stop token for any ToggleMTI source. Argument uses the JSON format: [{ "start": "START-TOKEN", "stop": "STOP-TOKEN" }] where 'START-TOKEN/END-TOKEN' are the corresponding start/stop tokens for toggling the trace plugins. Note that '\n' will be ignored if at start or end of the token. For additional information, use 'help' as the value of this parameter.

Type

string

Default value

uart_enable

Enable uart when the system starts up. (clock_rate and baud_rate are only valid when this option is enabled.)

Type

bool

Default value

0x0

unbuffered_output

Unbuffered output

Type

bool

Default value

0x0

untimed_fifos

Ignore the clock rate and transmit/receive serial data immediately

Type

bool

Default value

0x1

3.10.63 PL022_SSP

ARM PrimeCell Synchronous Serial Port(PL022). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1087: IP revisions support

Revision	Quality level
r1p4	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL022_SSP

This model has the following Iris instances:

Table 3-1088: PL022_SSP Iris instances

InstanceName	ComponentName
PL022_SSP	PL022_SSP
PL022_SSP.busslave	PVBusSlave
PL022_SSP.prescaler	ClockDivider

This model has the following MTI trace components:

Table 3-1089: PL022_SSP MTI instances

InstanceName	ComponentName
PL022_SSP.busslave	PVBusSlave
PL022_SSP.prescaler	ClockDivider

PL022_SSP contains the following CADI targets:

- ClockDivider
- PL022_SSP

Differences between the model and the RTL

Although the PL022_SSP component has clock input, it is not internally clock-driven. This is different to the hardware.



This component is a preliminary release. It is not a fully-supported peripheral.

Ports for PL022_SSP

Table 3-1090: Ports

Name	Protocol	Type	Description
clk	ClockSignal	Slave	Main PrimeCell SSP clock input.
clkkin	ClockSignal	Slave	PrimeCell SSP clock input.
clkout	ClockSignal	Master	Clock output.
intr	Signal	Master	Interrupt signaling.
pdbus	PVBus	Slave	Slave port for connection to PV bus master/decoder.
rorintr	Signal	Master	Receive overrun interrupt.
rtintr	Signal	Master	Receive timeout interrupt. We don't implement time out interrupt.
rx_dma_port	PL080_DMAC_DmaPortProtocol	Master	PrimeCell SSP receive DMA port.
rxdata	Value	Slave	PrimeCell SSP receive data.
rxintr	Signal	Master	Receive FIFO service request port.
tx_dma_port	PL080_DMAC_DmaPortProtocol	Master	PrimeCell SSP transmit DMA port.
txdata	Value	Master	PrimeCell SSP transmit data.

Name	Protocol	Type	Description
txintr	Signal	Master	Transmit FIFO service request.

Parameters for PL022_SSP

prescaler.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

prescaler.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

3.10.64 PL030_RTC

ARM PrimeCell Real Time Clock(PL030). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1091: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL030_RTC

This model has the following Iris instances:

Table 3-1092: PL030_RTC Iris instances

InstanceName	ComponentName
PL030_RTC	PL030_RTC
PL030_RTC.busslave	PVBusSlave
PL030_RTC.timer	ClockTimerThread
PL030_RTC.timer.timer	ClockTimerThread64
PL030_RTC.timer.timer.thread	SchedulerThread
PL030_RTC.timer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-1093: PL030_RTC MTI instances

InstanceName	ComponentName
PL030_RTC.busslave	PVBusSlave

PL030_RTC contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- PL030_RTC
- SchedulerThread
- SchedulerThreadEvent

Ports for PL030_RTC

Table 3-1094: Ports

Name	Protocol	Type	Description
clock	ClockSignal	Slave	Clock input, typically 1MHz, driving master count rate.
intr	Signal	Master	Interrupt signaling.
pvbus	PVBus	Slave	Slave port for connection to PV bus master/decoder.

3.10.65 PL031_RTC

ARM PrimeCell Real Time Clock(PL031). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1095: IP revisions support

Revision	Quality level
r1p3	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL031_RTC

This model has the following Iris instances:

Table 3-1096: PL031_RTC Iris instances

InstanceName	ComponentName
PL031_RTC	PL031_RTC
PL031_RTC.busslave	PVBusSlave
PL031_RTC.timer	ClockTimerThread
PL031_RTC.timer.timer	ClockTimerThread64

InstanceName	ComponentName
PL031_RTC.timer.timer.thread	SchedulerThread
PL031_RTC.timer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-1097: PL031_RTC MTI instances

InstanceName	ComponentName
PL031_RTC.busslave	PVBusSlave

PL031_RTC contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- PL031_RTC
- SchedulerThread
- SchedulerThreadEvent

About PL031_RTC

This component can provide a basic alarm function or long time base counter.

It has no impact on the performance of a PV system when idle or counting down. The component only executes code when the counter expires or during bus accesses.

Ports for PL031_RTC

Table 3-1098: Ports

Name	Protocol	Type	Description
clock	ClockSignal	Slave	Clock input, typically 1MHz, driving master count rate.
intr	Signal	Master	Interrupt signaling.
pvbust	PVBus	Slave	Slave port for connection to PV bus master/decoder.

Parameters for PL031_RTC

RTCDR_reset_value

Reset value for RTCDR

Type

int

Default value

0x0

RTCDR_use_current_time

Use current Unix/POSIX time for reset value for RTCDR. If true RTCDR_reset_value is ignored

Type

bool

Default value

0x1

3.10.66 PL041_AACI

ARM PrimeCell Advanced Audio CODEC Interface(PL041). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1099: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL041_AACI

This model has the following Iris instances:

Table 3-1100: PL041_AACI Iris instances

InstanceName	ComponentName
PL041_AACI	PL041_AACI
PL041_AACI.busslave	PVBusSlave
PL041_AACI.timer	ClockTimerThread
PL041_AACI.timer.timer	ClockTimerThread64
PL041_AACI.timer.timer.thread	SchedulerThread
PL041_AACI.timer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-1101: PL041_AACI MTI instances

InstanceName	ComponentName
PL041_AACI.busslave	PVBusSlave

PL041_AACI contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- PL041_AACI
- SchedulerThread
- SchedulerThreadEvent

About PL041_AACI

The PL041_AACI component is designed to connect to an audio output component such as [AudioOut_File](#) or [AudioOut_SDL](#).

The ability to play audio through this component depends on the AudioOut component in use and on the performance requirements of the software running on the simulated system. The rate of FIFO draining is controlled by the audio output to which the component is connected. This might not correspond to the rate that would be expected from the reference clock.

This component also contains a minimal register model of the LM4529 secondary codec as implemented on development boards supplied by Arm.



This component is not a complete implementation of the AACI because the following functionality is not implemented:

- Audio input
- DMA access to FIFOs, rather than Programmed I/O
- Programming of the secondary codec through FIFOs rather than slot registers

Ports for PL041_AACI

Table 3-1102: Ports

Name	Protocol	Type	Description
audio	AudioControl	Master	Used to communicate with an audio out device.
clk_in_ref	ClockSignal	Slave	Reference clock input, typically 25MH.
dma_rx	PL080_DMAC_DmaPortProtocol	Master	DMA receive port.
dma_tx	PL080_DMAC_DmaPortProtocol	Master	DMA transmit port.
irq	Signal	Master	Single IRQ output port.
pvbuss	PVBus	Slave	Slave port for connection to PV bus master/decoder.

Parameters for PL041_AACI

enabled
Host interface connection enabled

Type
bool

Default value
0x1

3.10.67 PL050_KMI

ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1103: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL050_KMI

This model has the following Iris instances:

Table 3-1104: PL050_KMI Iris instances

InstanceName	ComponentName
PL050_KMI	PL050_KMI
PL050_KMI.busslave	PVBusSlave
PL050_KMI.clk_divider	ClockDivider

This model has the following MTI trace components:

Table 3-1105: PL050_KMI MTI instances

InstanceName	ComponentName
PL050_KMI.busslave	PVBusSlave
PL050_KMI.clk_divider	ClockDivider

PL050_KMI contains the following CADI targets:

- ClockDivider
- PL050_KMI

About PL050_KMI

This model communicates with models of PS/2-like devices, for example a PS2Keyboard or PS2Mouse.

Ports for PL050_KMI

Table 3-1106: Ports

Name	Protocol	Type	Description
clock	ClockSignal	Slave	Clock input, typically 1MHz, which sets the master transmit/receive rate.
intr	Signal	Master	Master port signaling completion of transmit or receive.
ps2device	PS2Data	Slave	Used to communicate with a PS/2-like device.
pvbus	PVBus	Slave	Slave port for connection to PV bus master/decoder.

Parameters for PL050_KMI

clk_divider.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

`clk_divider.mul`

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

3.10.68 PL061_GPIO

ARM PrimeCell General Purpose Input/Output(PL061). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1107: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL061_GPIO

This model has the following Iris instances:

Table 3-1108: PL061_GPIO Iris instances

InstanceName	ComponentName
PL061_GPIO	PL061_GPIO
PL061_GPIO.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-1109: PL061_GPIO MTI instances

InstanceName	ComponentName
PL061_GPIO.busslave	PVBusSlave

PL061_GPIO contains the following CADI targets:

- [PL061_GPIO](#)

About PL061_GPIO

This component provides eight programmable inputs or outputs. Ports of different widths can be created by multiple instantiation. In addition, an interrupt interface is provided to configure any number of pins as interrupt sources.

Ports for PL061_GPIO

Table 3-1110: Ports

Name	Protocol	Type	Description
GPIO_In	Value	Slave	Input lines. 32-bit data in, only [7:0] is used.
GPIO_Intr	Signal	Master	Interrupt signal indicating to an interrupt controller that an interrupt occurred in one or more of the GPIO_In lines.
GPIO_MIS	Value	Master	Indicates the masked interrupt status. 32-bit data out , only [7:0] is used. NOT necessary, as the GPIOMIS can be read from address 0x418.
GPIO_Out	Value	Master	Output lines. 32-bit data out, only [7:0] is used.
pvbuss	PVBus	Slave	Slave port for register access.

Parameters for PL061_GPIO

init_inputs

Default input values [7:0]

Type

int

Default value

0x0

3.10.69 PL080_DMAC

ARM PrimeCell DMA Controller(PL080/081). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1111: IP revisions support

Revision	Quality level
r1p3	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL080_DMAC

This model has the following Iris instances:

Table 3-1112: PL080_DMAC Iris instances

InstanceName	ComponentName
PL080_DMAC	PL080_DMAC

InstanceName	ComponentName
PL080_DMAC.busmaster0	PVBusMaster
PL080_DMAC.busmaster1	PVBusMaster
PL080_DMAC.busslave	PVBusSlave
PL080_DMAC.timer	ClockTimerThread
PL080_DMAC.timer.timer	ClockTimerThread64
PL080_DMAC.timer.timer.thread	SchedulerThread
PL080_DMAC.timer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-1113: PL080_DMAC MTI instances

InstanceName	ComponentName
PL080_DMAC.busmaster0	PVBusMaster
PL080_DMAC.busmaster1	PVBusMaster
PL080_DMAC.busslave	PVBusSlave

PL080_DMAC contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- PL080_DMAC
- SchedulerThread
- SchedulerThreadEvent

About PL080_DMAC

This component provides 8 configurable DMA channels and 16 DMA ports for handshaking with peripherals. You can configure each channel to operate in one of eight flow control modes either under DMA control or the control of the source or destination peripheral. Transfers can occur on either master channel and can optionally be endian converted on both source and destination transfers.

This component might have a significant impact on system performance in certain flow control modes. Channels configured for small bursts, or using single bursts, and with peripheral DMA handshaking could add significant overhead. The peripheral has not been fully optimized to make use of the advanced features of the PVBus model.

Ports for PL080_DMAC

Table 3-1114: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Clock signal to control DMA transfer rate.
dma_port[16]	PL080_DMAC_DmaPortProtocol	Slave	request/response ports for communicating with devices.
interr	Signal	Master	DMA error interrupt signal.

Name	Protocol	Type	Description
intr	Signal	Master	Combined DMA error and terminal count signal.
inttc	Signal	Master	DMA terminal count signal.
pdbus0_m	PVBus	Master	Master bus interface 0 for DMA transfers.
pdbus1_m	PVBus	Master	Master bus interface 1 for DMA transfers.
pdbus_s	PVBus	Slave	Slave port for register accesses.
reset_in	Signal	Slave	System reset.

Parameters for PL080_DMAC

activate_delay

request delay

Type

int

Default value

0x0

fifo_size

Channel FIFO size in bytes

Type

int

Default value

0x10

generate_clear

Generate clear response

Type

bool

Default value

0x0

max_transfer

Largest atomic transfer

Type

int

Default value

0x100

3.10.70 PL110_CLCD

ARM PrimeCell Color LCD Controller(PL110). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1115: IP revisions support

Revision	Quality level
r1p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL110_CLCD

This model has the following Iris instances:

Table 3-1116: PL110_CLCD Iris instances

InstanceName	ComponentName
PL110_CLCD	PL110_CLCD
PL110_CLCD.pl11x_clcd	PL11x_CLCD
PL110_CLCD.pl11x_clcd.busmaster	PVBusMaster
PL110_CLCD.pl11x_clcd.busslave	PVBusSlave
PL110_CLCD.pl11x_clcd.timer	ClockTimerThread
PL110_CLCD.pl11x_clcd.timer.timer	ClockTimerThread64
PL110_CLCD.pl11x_clcd.timer.timer.thread	SchedulerThread
PL110_CLCD.pl11x_clcd.timer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-1117: PL110_CLCD MTI instances

InstanceName	ComponentName
PL110_CLCD.pl11x_clcd	PL11x_CLCD
PL110_CLCD.pl11x_clcd.busmaster	PVBusMaster
PL110_CLCD.pl11x_clcd.busslave	PVBusSlave

PL110_CLCD contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- PL110_CLCD
- PL11x_CLCD
- SchedulerThread
- SchedulerThreadEvent

About PL110_CLCD

This implementation provides a register model of the LCD controller.

You can connect the model through a framebuffer port to a visualization component, for example, so that LCD output can be viewed.

The implementation is optimized for situations where the majority of the framebuffer does not change. For instance, displaying full-screen video results in significantly reduced performance. Rendering pixel data into an appropriate form for the framebuffer port (rasterization) can also take a significant amount of simulation time. If the pixel data are coming from a PVBUSSlave region that has been configured as memory-like, rasterization only occurs in regions where memory contents are modified.

See also [1.14 Visualisation library](#) on page 59.

Ports for PL110_CLCD

Table 3-1118: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Master clock input, typically 24MHz, to drive pixel clock timing.
control	Value	Slave	Auxiliary control register 1.
display	LCD	Master	Connection to visualization component.
intr	Signal	Master	Interrupt signaling for flyback events.
pvbuss	PVBUS	Slave	Slave port for register access.
pvbuss_m	PVBUS	Master	DMA port for video data.

Parameters for PL110_CLCD

disable_snooping_dma

Disable DMA snooping

Type

bool

Default value

0x0

pixel_double_limit

Minimum LCD pixel width before display will be zoomed

Type

int

Default value

0x12c

3.10.71 PL111_CLCD

ARM PrimeCell Color LCD Controller(PL111). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1119: IP revisions support

Revision	Quality level
r0p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL111_CLCD

This model has the following Iris instances:

Table 3-1120: PL111_CLCD Iris instances

InstanceName	ComponentName
PL111_CLCD	PL111_CLCD
PL111_CLCD.pl11x_clcd	PL11x_CLCD
PL111_CLCD.pl11x_clcd.busmaster	PVBusMaster
PL111_CLCD.pl11x_clcd.busslave	PVBusSlave
PL111_CLCD.pl11x_clcd.timer	ClockTimerThread
PL111_CLCD.pl11x_clcd.timer.timer	ClockTimerThread64
PL111_CLCD.pl11x_clcd.timer.timer.thread	SchedulerThread
PL111_CLCD.pl11x_clcd.timer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-1121: PL111_CLCD MTI instances

InstanceName	ComponentName
PL111_CLCD.pl11x_clcd	PL11x_CLCD
PL111_CLCD.pl11x_clcd.busmaster	PVBusMaster
PL111_CLCD.pl11x_clcd.busslave	PVBusSlave

PL111_CLCD contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- PL111_CLCD
- PL11x_CLCD
- SchedulerThread
- SchedulerThreadEvent

About PL111_CLCD

This component implements the hardware cursor support of the PL111_CLCD, which is the main difference with PL110_CLCD.

See also [1.14 Visualisation library](#) on page 59.

Ports for PL111_CLCD

Table 3-1122: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Master clock input, typically 24MHz, to drive pixel clock timing.
control	Value	Slave	Auxiliary control register 1.
display	LCD	Master	Connection to visualization component.
intr	Signal	Master	Interrupt signaling for flyback events.
pvbust	PVBus	Slave	Slave port for register access.
pvbust_m	PVBus	Master	DMA port for video data.

Parameters for PL111_CLCD

disable_snooping_dma

Disable DMA snooping

Type

bool

Default value

0x0

pixel_double_limit

Minimum LCD pixel width before display will be zoomed

Type

int

Default value

0x12c

3.10.72 PL180_MCI

ARM PrimeCell Multimedia Card Interface (PL180). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1123: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL180_MCI

This model has the following Iris instances:

Table 3-1124: PL180_MCI Iris instances

InstanceName	ComponentName
PL180_MCI	PL180_MCI
PL180_MCI.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-1125: PL180_MCI MTI instances

InstanceName	ComponentName
PL180_MCI.busslave	PVBusSlave

PL180_MCI contains the following CADI targets:

- PL180_MCI

About PL180_MCI

When paired with an MMC card model, the PL180_MCI component provides emulation of a flexible, persistent storage mechanism. See [3.10.49 MMC](#) on page 3768. The PL180_MCI component fully models the registers of the corresponding PrimeCell, but supports a subset of the functionality of the PL180:

- The controller supports block mode transfers, but does not currently support streaming data transfer.
- The controller can be attached to a single MMC device. The MMC bus mode and SDIO modes of the PL180 PrimeCell are not supported.
- Command and Data timeouts are not simulated.
- Payload CRC errors are not simulated.
- The DMA interface present in the PL180 PrimeCell is not modeled.
- Minimal timing is implemented within the model.



Note

At compile time, you can enable command tracing within the PL180_MCI component by modifying the `PL180_TRACE` macro in the `MMC.lisa` file. This sends command and event trace to standard output. You can use this output to help diagnose device driver and controller-to-card protocol issues.

Ports for PL180_MCI

Table 3-1126: Ports

Name	Protocol	Type	Description
MCIINTR[2]	Signal	Master	Interrupts.
mmc_m	MMC_Protocol	Master	The MultiMediaCard (MMC) master port.

Name	Protocol	Type	Description
pvbuss	PVBus	Slave	Slave port for register access.

Parameters for PL180_MCI

p1180_fifo_depth

PL180 FIFO Depth

Type

int

Default value

0x10

3.10.73 PL192_VIC

ARM PrimeCell Vectored Interrupt Controller(PL192). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1127: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL192_VIC

This model has the following Iris instances:

Table 3-1128: PL192_VIC Iris instances

InstanceName	ComponentName
PL192_VIC	PL192_VIC
PL192_VIC.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-1129: PL192_VIC MTI instances

InstanceName	ComponentName
PL192_VIC.busslave	PVBusSlave

PL192_VIC contains the following CADI targets:

- [PL192_VIC](#)

About PL192_VIC

This component aggregates interrupts and generates interrupt signals for the Arm® processor. When coupled with an Arm processor that provides a VIC port, routing to the appropriate interrupt

handler can optionally be performed in hardware, reducing interrupt latency. The PL192_VIC can also be daisy-chained with other PL192 VICs to permit more than 32 interrupts. The VIC supports hardware and software prioritization of interrupts.

Ports for PL192_VIC

Table 3-1130: Ports

Name	Protocol	Type	Description
nVICFIQ	Signal	Master	Send out FIQ signal to the next level VIC or CPI.
nVICFIQIN	Signal	Slave	Used to receive FIQ signal when daisy chained.
nVICIRQ	Signal	Master	Send out IRQ signal to the next level VIC or procesessor.
nVICIRQIN	Signal	Slave	Used to receive IRQ signal when daisy chained.
pvbus	PVBus	Slave	Slave port for register access.
VICIntSource[32]	Signal	Slave	Interrupt source input sources.
VICIRQACK	Signal	Slave	Receive acknowledge signal from next level VIC or processor.
VICIRQACKOUT	Signal	Master	Used to send out acknowledge signals when daisy chained.
VICVECTADDRIN	ValueState	Slave	Used to receive vector address when daisy chained.
VICVECTADDROUT	ValueState	Master	Used to send vector address to next level VIC or processor.

3.10.74 PL310_L2CC

ARM PrimeCell Level 2 Cache Controller (PL310). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1131: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL310_L2CC

This model has the following Iris instances:

Table 3-1132: PL310_L2CC Iris instances

InstanceName	ComponentName
PL310_L2CC	PL310_L2CC

This model has the following MTI trace components:

Table 3-1133: PL310_L2CC MTI instances

InstanceName	ComponentName
PL310_L2CC	PL310_L2CC

PL310_L2CC contains the following CADI targets:

- PL310_L2CC

About PL310_L2CC

The presence of additional on-chip secondary cache can improve performance when significant memory traffic is generated by the processor. A secondary cache assumes the existence of a Level 1, or primary, cache that is closely coupled or internal to the processor.

This component has two modes of operation, which are controlled by the `cache-state_modelled` parameter:

Register view

Cache control registers are present but the cache behavior is not modeled.

Functional model

Cache behavior is modeled.

Arm supports the use of the PL310 when connected to the Arm® Cortex®-A5 or Cortex-A9 processor.

This component implements the programmer-visible functionality of the PL310, and excludes some non-programmer visible features. The following features are implemented in the model:

- Physically addressed and physically tagged.
- Lockdown format C supported, for data and instructions. Lockdown format C is also known as way locking.
- Lockdown by line supported.
- Lockdown by master ID supported.
- Direct mapped to 16-way associativity, depending on the configuration and the use of lockdown registers. The associativity is configurable as 8 or 16.
- L2 cache available size can be 16KB to 8MB, depending on configuration and the use of the lockdown registers.
- Fixed line length of 32 bytes (8 words or 256 bits).
- Supports all of the AXI cache modes:
 - write-through and write-back.
 - read allocate, write allocate, read and write allocate.
- Force write-allocate option to always have cacheable writes allocated to L2 cache, for processors not supporting this mode.
- Normal memory non-cacheable shared reads are treated as cacheable non-allocatable. Normal memory non-cacheable shared writes are treated as cacheable write-through no write-allocate. There is an option, Shared Override, to override this behavior.
- TrustZone® support, with the following features:
 - Non-Secure (NS) tag bit added in tag RAM and used for lookup in the same way as an address bit.

- NS bit in Tag RAM used to determine security level of evictions to L3.
- Restrictions for NS accesses for control, configuration, and maintenance registers to restrict access to secure data.
- Pseudo-Random victim selection policy. You can make this deterministic with use of lockdown registers.
- Software option to enable exclusive cache configuration.
- Configuration registers accessible using address decoding in the component.
- Interrupt triggering in case of an error response when accessing L3.
- Maintenance operations.
- Prefetching capability.

The performance of this component depends on the configuration of the associated L1 caches and the mode it is in:

- Register mode: no significant effect.
- Functional mode with functional-mode L1: the addition of a functional L2 cache has minimal further impact on performance when running applications that are cache-bound.
- Functional mode with a register-mode L1: there is a significant impact on system performance.



Setting timing delays in this model does not impact the simulation speed. Generally, timing delays are only modeled for CPUs.

See also [1.5.2 Caches in PV models](#) on page 29.

Differences between the model and the RTL

This model does not implement the following features, most of which are not relevant from a PV modeling point of view:

- There is no interface to the data and tag RAM as they are embedded to the model.
- Critical word first linefill is not supported, as it is not relevant for PV modeling.
- Buffers are not modeled.
- Outstanding accesses on slave and master ports cannot occur by design in a PV model as all transactions are atomic.
- Option to select one or two master ports and option to select one or two slave ports is not supported. Only one master port and one slave port is supported.
- Clock management and power modes are not supported, as they are not relevant for PV modeling.
- Wait, latency, clock enable, parity, and error support for data and tag RAMs not included, as they are not relevant for PV modeling, and the data and tag RAMs embedded in the model cannot generate error responses.

- MBIST support is not included.
- Debug mode and debug registers are not supported.
- Test mode and scan chains are not supported.
- L2 cache event monitoring is not supported.
- Address filtering in the master ports is not supported.
- Performance counters are not supported.
- Specific Cortex-A9 related optimizations are not supported: Prefetch hints, Full line of zero and Early write response.
- Hazard detection is not required because of the atomic nature of the accesses at PV modeling and the fact that buffers are not modeled, therefore hazards cannot occur.
- Registers that belong to unimplemented features are accessible but do not have any functionality.

This model implements the following features differently to the hardware:

- Error handling. DECERR from the master port is mapped to SLVERR. Internal errors in cache RAM (like parity errors) cannot happen in the model.
- Background cache operations do not occur in the background. They occur atomically.
- The LOCKDOWN_BY_LINE and LOCKDOWN_BY_MASTER parameter values are reflected in the CacheType register, but the feature is not switched off when the parameter is 0.

This feature is additional:

- Data RAM and Tag RAM are embedded to the model.

Ports for PL310_L2CC

Table 3-1134: Ports

Name	Protocol	Type	Description
DECERRINTR	Signal	Master	Decode error received on master port from L3.
ECNTRINTR	Signal	Master	Event Counter Overflow / Increment.
ERRRDINTR	Signal	Master	Error on L2 data RAM read.
ERRRTINTR	Signal	Master	Error on L2 tag RAM read.
ERRWDINTR	Signal	Master	Error on L2 data RAM write.
ERRWTINTR	Signal	Master	Error on L2 tag RAM write.
L2CCINTR	Signal	Master	Combined interrupt output.
PARRDINTR	Signal	Master	Parity error on L2 data RAM read.
PARRTINTR	Signal	Master	Parity error on L2 tag RAM read.
pvbus_m	PVBus	Master	Master port for connection to PV bus master/decoder.
pvbus_s	PVBus	Slave	Slave port for connection to PV bus master/decoder.
SLVERRINTR	Signal	Master	Slave error on master port from L3.

Parameters for PL310_L2CC

ASSOCIATIVITY

Associativity for Auxiliary Control Register

Type

int

Default value

0x0

CACHEID

Cache controller cache ID

Type

int

Default value

0x0

CFGBIGEND

Big-endian mode for accessing configuration registers out of reset

Type

int

Default value

0x0

LOCKDOWN_BY_LINE

Lockdown by line - value is reflected in CacheType register Bit 25, but the feature is not switched off when the parameter is 0

Type

int

Default value

0x0

LOCKDOWN_BY_MASTER

Lockdown by master - value is reflected in CacheType register Bit 26, but the feature is not switched off when the parameter is 0

Type

int

Default value

0x0

REGFILEBASE

Base address for accessing configuration registers

Type

int

Default value

0x1f002000

WAYSIZE

Size of ways for Auxiliary Control Register

Type

int

Default value

0x1

cache-state_modelled

Specifies whether real cache state is modelled (vs. register model)

Type

bool

Default value

0x0

delay_cache_hit

Cost to handle a cache hit

Type

int

Default value

0x0

delay_cache_miss

Cost to handle a cache miss

Type

int

Default value

0x0

delay_cache_perbeat

Cost to handle one beat of cache data movement

Type

int

Default value

0x0

3.10.75 PL330_DMAC

ARM PrimeCell DMA Controller(PL330). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1135: IP revisions support

Revision	Quality level
rOp0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL330_DMAC

This model has the following Iris instances:

Table 3-1136: PL330_DMAC Iris instances

InstanceName	ComponentName
PL330_DMAC	PL330_DMAC
PL330_DMAC.busmaster	PVBusMaster
PL330_DMAC.busslave	PVBusSlave
PL330_DMAC.busslave_ns	PVBusSlave
PL330_DMAC.timer	ClockTimerThread
PL330_DMAC.timer.timer	ClockTimerThread64
PL330_DMAC.timer.timer.thread	SchedulerThread
PL330_DMAC.timer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-1137: PL330_DMAC MTI instances

InstanceName	ComponentName
PL330_DMAC	PL330_DMAC
PL330_DMAC.busmaster	PVBusMaster
PL330_DMAC.busslave	PVBusSlave
PL330_DMAC.busslave_ns	PVBusSlave

PL330_DMAC contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- PL330_DMAC
- SchedulerThread
- SchedulerThreadEvent

About PL330_DMAC

The model uses a single LISA+ component but with a C++ model for each of the channels included in the LISA+ file. Enabled channels are kept on an enabled channels stack in priority order. When a channel state changes, re-arbitration takes place to make the highest (topmost) channel active.

Each transaction carries the identity of the requesting thread. This controller has up to eight channel threads and a manager thread. Each has an ID. In the hardware, the ID is $AxID[3:0]$, with $0x0 - (\text{numberOfChannels} - 1)$ identifying channels and $(\text{numberOfChannels})$ identifying the manager. For example, $0x0-0x7$ and $0x8$, respectively. The manager originates only instruction fetches, and the manager ID is also used for instruction fetches issued by the channels.

In the model, the identity of the requesting thread is encoded into each transaction using the low-order 16 bits of the Master ID field:

- Channel data: $0-7$.
- Channel instruction fetch: $0xffff$.
- Manager instruction fetch: $0xffff$.

If a downstream component needs to know the IDs of bus masters that use either the low-order 16 bits or the label, use the label. The `LabellerForDMA330` component shifts the low-order 16 bits into the label, while providing a degree of control over the label encoding. The example below maintains separate IDs for each data channel while using the correct hardware ID to identify instruction fetch for a DMA-330 with 8 channels:

```
pl330_dma : PL330_DMAC( "p_max_channels" = 8 );
dma_labeller : LabellerForDMA330(
    "dma330_discriminate_data_channels" = true,
    "dma330_s_instruction_label" = 8,
    "dma330_ns_instruction_label" = 8 );
pl330_dma.pvbus_m => dma_labeller.pvbus_s;
dma_labeller.pvbus_m => output_bus.pvbus_s;
```

Ports for PL330_DMAC

Table 3-1138: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Clock input.
irq_abort_master_port	Signal	Master	Undefined instruction or instruction error.
irq_master_port[32]	Signal	Master	Sets when DMASEV.
pvbus_m	PVBus	Master	Master port for all memory accesses.
pvbus_s	PVBus	Slave	Slave port for all register accesses (secure).
pvbus_s_ns	PVBus	Slave	Slave port for all register accesses (non-secure).
reset_in	Signal	Slave	System reset.

Parameters for PL330_DMAC

activate_delay

request delay

Type

int

Default value

0x0

fifo_size

Channel FIFO size in bytes

Type

int

Default value

0x10

generate_clear

Generate clear response

Type

bool

Default value

0x0

max_transfer

Largest atomic transfer

Type

int

Default value

0x100

p_axi_bus_width_param

AXI bus width

Type

int

Default value

0x20

p_buffer_depth

buffer depth

Type

int

Default value

0x10

p_cache_line_words

number of words in a cache line

Type

int

Default value

0x1

p_cache_lines

number of cache lines

Type

int

Default value

0x1

p_controller_boots

DMA boots from reset

Type

bool

Default value

0x1

p_controller_nsecure

Controller non-secure at reset (boot_manager_ns)

Type

bool

Default value

0x0

p_irq_nsecure

Interrupts non-secure at reset

Type

int

Default value

0x0

p_lsq_read_size

LSQ read buffer depth

Type

int

Default value

0x4

p_lsq_write_size

LSQ write buffer depth

Type

int

Default value

0x4

p_max_channels

virtual channels

Type

int

Default value

0x8

p_max_irqs

number of interrupts

Type

int

Default value

0x20

p_max_periph

number of peripheral interfaces

Type

int

Default value

0x20

p_perip_request_acceptance_0

Peripheral 0 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_1

Peripheral 1 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_10
Peripheral 10 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_11
Peripheral 11 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_12
Peripheral 12 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_13
Peripheral 13 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_14
Peripheral 14 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_15
Peripheral 15 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_16

Peripheral 16 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_17

Peripheral 17 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_18

Peripheral 18 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_19

Peripheral 19 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_2

Peripheral 2 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_20

Peripheral 20 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_21
Peripheral 21 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_22
Peripheral 22 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_23
Peripheral 23 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_24
Peripheral 24 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_25
Peripheral 25 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_26
Peripheral 26 request acceptance
Type
int
Default value
0x2

p_perip_request_acceptance_27

Peripheral 27 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_28

Peripheral 28 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_29

Peripheral 29 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_3

Peripheral 3 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_30

Peripheral 30 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_31

Peripheral 31 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_4

Peripheral 4 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_5

Peripheral 5 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_6

Peripheral 6 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_7

Peripheral 7 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_8

Peripheral 8 request acceptance

Type

int

Default value

0x2

p_perip_request_acceptance_9

Peripheral 9 request acceptance

Type

int

Default value

0x2

p_periph_nsecure

Peripherals non-secure at reset

Type

bool

Default value

0x0

p_read_issuing_capability

AXI read issuing capability

Type

int

Default value

0x1

p_reset_pc

DMA PC at reset

Type

int

Default value

0x60000000

p_write_issuing_capability

AXI write issuing capability

Type

int

Default value

0x1

revision

revision ID

Type

string

Default value

r0p0

3.10.76 PL340_DMC

ARM PrimeCell Dynamic Memory Controller(PL340). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1139: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL340_DMC

This model has the following Iris instances:

Table 3-1140: PL340_DMC Iris instances

InstanceName	ComponentName
PL340_DMC	PL340_DMC
PL340_DMC.apb_slave	PVBusSlave
PL340_DMC.exclusive_monitor0	PVBusExclusiveMonitor
PL340_DMC.exclusive_monitor0.bus_mapper	PVBusMapper
PL340_DMC.exclusive_monitor1	PVBusExclusiveMonitor
PL340_DMC.exclusive_monitor1.bus_mapper	PVBusMapper
PL340_DMC.exclusive_monitor2	PVBusExclusiveMonitor
PL340_DMC.exclusive_monitor2.bus_mapper	PVBusMapper
PL340_DMC.exclusive_monitor3	PVBusExclusiveMonitor
PL340_DMC.exclusive_monitor3.bus_mapper	PVBusMapper

This model has the following MTI trace components:

Table 3-1141: PL340_DMC MTI instances

InstanceName	ComponentName
PL340_DMC.apb_slave	PVBusSlave
PL340_DMC.exclusive_monitor0	PVBusExclusiveMonitor
PL340_DMC.exclusive_monitor0.bus_mapper	PVBusMapper
PL340_DMC.exclusive_monitor1	PVBusExclusiveMonitor
PL340_DMC.exclusive_monitor1.bus_mapper	PVBusMapper
PL340_DMC.exclusive_monitor2	PVBusExclusiveMonitor
PL340_DMC.exclusive_monitor2.bus_mapper	PVBusMapper
PL340_DMC.exclusive_monitor3	PVBusExclusiveMonitor
PL340_DMC.exclusive_monitor3.bus_mapper	PVBusMapper

PL340_DMC contains the following CADI targets:

- PL340_DMC
- PVBusExclusiveMonitor

About PL340_DMC

This component provides an interface for up to four DRAM chips. The implementation also provides an APB interface to configure the controller behavior. You can access the registers through the APB interface.

Ports for PL340_DMC

Table 3-1142: Ports

Name	Protocol	Type	Description
apb_interface	PVBus	Slave	Receive the apb config read/writes here.
axi_if_in[4]	PVBus	Slave	Receive the axi reads/writes here; up to four chips can be connected.
axi_if_out[4]	PVBus	Master	The output ports where the actual mem chips are connected.

Parameters for PL340_DMC

IF_CHIP0
Set this parameter to 0 if memory is connected
Type
int
Default value
0xffffffffffffffff

IF_CHIP1
Set this parameter to 0 if memory is connected
Type
int
Default value
0xffffffffffffffff

IF_CHIP2
Set this parameter to 0 if memory is connected
Type
int
Default value
0xffffffffffffffff

IF_CHIP3
Set this parameter to 0 if memory is connected
Type
int
Default value
0xffffffffffffffff

MEMORY_WIDTH

Set this parameter to 0 if memory is connected

Type

int

Default value

0x20

exclusive_monitor0.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor0.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor0.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor0.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor0.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor0.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor0.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor0.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor0.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

exclusive_monitor1.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor1.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor1.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor1.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor1.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor1.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor1.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor1.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor1.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

exclusive_monitor2.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor2.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor2.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor2.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor2.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor2.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor2.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor2.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor2.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

exclusive_monitor3.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor3.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor3.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor3.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor3.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor3.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor3.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor3.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor3.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

3.10.77 PL350_SMC

ARM PrimeCell Static Memory Controller(PL350). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1143: IP revisions support

Revision	Quality level
r1p2	Full support
r2p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL350_SMC

This model has the following Iris instances:

Table 3-1144: PL350_SMC Iris instances

InstanceName	ComponentName
PL350_SMC	PL350_SMC
PL350_SMC.addr_remapper	TZSwitch
PL350_SMC.addr_remapper.pvbus_mapper	PVBusMapper
PL350_SMC.apb_slave	PVBusSlave

InstanceName	ComponentName
PL350_SMC.exclusive_monitor0_0	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor0_0.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor0_1	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor0_1.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor0_2	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor0_2.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor0_3	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor0_3.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor1_0	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor1_0.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor1_1	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor1_1.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor1_2	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor1_2.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor1_3	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor1_3.bus_mapper	PVBusMapper
PL350_SMC.master_if0_0	PVBusMaster
PL350_SMC.master_if0_1	PVBusMaster
PL350_SMC.master_if0_2	PVBusMaster
PL350_SMC.master_if0_3	PVBusMaster
PL350_SMC.master_if1_0	PVBusMaster
PL350_SMC.master_if1_1	PVBusMaster
PL350_SMC.master_if1_2	PVBusMaster
PL350_SMC.master_if1_3	PVBusMaster
PL350_SMC.nand_remap_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-1145: PL350_SMC MTI instances

InstanceName	ComponentName
PL350_SMC.addr_remapper.pvbus_mapper	PVBusMapper
PL350_SMC.apb_slave	PVBusSlave
PL350_SMC.exclusive_monitor0_0	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor0_0.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor0_1	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor0_1.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor0_2	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor0_2.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor0_3	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor0_3.bus_mapper	PVBusMapper

InstanceName	ComponentName
PL350_SMC.exclusive_monitor1_0	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor1_0.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor1_1	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor1_1.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor1_2	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor1_2.bus_mapper	PVBusMapper
PL350_SMC.exclusive_monitor1_3	PVBusExclusiveMonitor
PL350_SMC.exclusive_monitor1_3.bus_mapper	PVBusMapper
PL350_SMC.master_if0_0	PVBusMaster
PL350_SMC.master_if0_1	PVBusMaster
PL350_SMC.master_if0_2	PVBusMaster
PL350_SMC.master_if0_3	PVBusMaster
PL350_SMC.master_if1_0	PVBusMaster
PL350_SMC.master_if1_1	PVBusMaster
PL350_SMC.master_if1_2	PVBusMaster
PL350_SMC.master_if1_3	PVBusMaster
PL350_SMC.nand_remap_slave	PVBusSlave

PL350_SMC contains the following CADI targets:

- PL350_SMC
- PVBusExclusiveMonitor
- TZSwitch

About PL350_SMC

This component provides two memory interfaces. Each interface can be connected to a maximum of four memory devices, giving a total of eight inputs from the PVBusDecoder and eight outputs to either SRAM or NAND devices. Only one kind of memory can be connected to a particular interface, either SRAM or NAND.

It provides a PVBus slave to control the device behavior. A remap port is also provided to assist in remapping particular memory regions.

This component is optimized to have negligible impact on transaction performance, except when memory remap settings are changed, when there might be a significant effect.

Ports for PL350_SMC

Table 3-1146: Ports

Name	Protocol	Type	Description
apb_interface	PVBus	Slave	This is where we expect to receive all the APB data which is used to read/write the device regs.
axi_chip_if0_in[4]	PVBus	Slave	This is where we expect to receive all the AXI data which is used to read/wrie NAND/RAM mem.

Name	Protocol	Type	Description
axi_chip_if0_out[4]	PVBus	Master	Master interface 0 to connect to SRAM/NAND.
axi_chip_if1_in[4]	PVBus	Slave	This is where we expect to receive all the AXI data which is used to read/wrie NAND/RAM mem.
axi_chip_if1_out[4]	PVBus	Master	Master interface 1 to connect to SRAM/NAND.
axi_remap	PVBus	Slave	This is the remap port that the designer needs to connect to zero.
irq_in_if0	Signal	Slave	Interrupt signals from devices connected on interface 0.
irq_in_if1	Signal	Slave	Interrupt signals from device connected on interface 1.
irq_out	Signal	Master	Interrupt port.
nand_remap_port	PVBus	Slave	Remaps the connected NAND port to 0x0.

Parameters for PL350_SMC

IF0_CHIP0

Interface 0 chip 0 connected

Type

bool

Default value

0x0

IF0_CHIP0_BASE

Interface 0 chip 0 Base address

Type

int

Default value

0x0

IF0_CHIP0_SIZE

Interface 0 chip 0 Size

Type

int

Default value

0x0

IF0_CHIP1

Interface 0 chip 1 connected

Type

bool

Default value

0x0

IF0_CHIP1_BASE

Interface 0 chip 1 Base address

Type

int

Default value

0x0

IF0_CHIP1_SIZE

Interface 0 chip 1 Size

Type

int

Default value

0x0

IF0_CHIP2

Interface 0 chip 2 connected

Type

bool

Default value

0x0

IF0_CHIP2_BASE

Interface 0 chip 2 Base address

Type

int

Default value

0x0

IF0_CHIP2_SIZE

Interface 0 chip 2 Size

Type

int

Default value

0x0

IF0_CHIP3

Interface 0 chip 3 connected

Type

bool

Default value

0x0

IF0_CHIP3_BASE

Interface 0 chip 3 Base address

Type

int

Default value

0x0

IF0_CHIP3_SIZE

Interface 0 chip 3 Size

Type

int

Default value

0x0

IF0_MEM_TYPE_PARAMETER

Interface 0 Mem type

Type

int

Default value

0x0

IF1_CHIP0

Interface 1 chip 0 connected

Type

bool

Default value

0x0

IF1_CHIP0_BASE

Interface 1 chip 0 Base address

Type

int

Default value

0x0

IF1_CHIP0_SIZE

Interface 1 chip 0 Size

Type

int

Default value

0x0

IF1_CHIP1

Interface 1 chip 1 connected

Type

bool

Default value

0x0

IF1_CHIP1_BASE

Interface 1 chip 1 Base address

Type

int

Default value

0x0

IF1_CHIP1_SIZE

Interface 1 chip 1 Size

Type

int

Default value

0x0

IF1_CHIP2

Interface 1 chip 2 connected

Type

bool

Default value

0x0

IF1_CHIP2_BASE

Interface 1 chip 2 Base address

Type

int

Default value

0x0

IF1_CHIP2_SIZE

Interface 1 chip 2 Size

Type

int

Default value

0x0

IF1_CHIP3
Interface 1 chip 3 connected

Type
bool

Default value
0x0

IF1_CHIP3_BASE
Interface 1 chip 3 Base address

Type
int

Default value
0x0

IF1_CHIP3_SIZE
Interface 1 chip 3 Size

Type
int

Default value
0x0

IF1_MEM_TYPE_PARAMETER
Interface 1 Mem type

Type
int

Default value
0x0

PERIPH_ID_0
Periph_ID_0 value

Type
int

Default value
0x52

REMAP
Remap the device

Type
int

Default value
0xffffffffffffffff

addr_remapper.normal

Normal Port

Type

int

Default value

0x2

addr_remapper.secure

Secure Port

Type

int

Default value

0x1

exclusive_monitor0_0.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor0_0.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor0_0.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor0_0.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor0_0.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor0_0.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor0_0.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor0_0.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor0_0.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

exclusive_monitor0_1.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor0_1.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor0_1.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor0_1.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor0_1.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor0_1.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor0_1.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor0_1.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor0_1.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

exclusive_monitor0_2.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor0_2.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor0_2.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor0_2.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor0_2.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor0_2.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor0_2.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor0_2.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor0_2.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

exclusive_monitor0_3.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor0_3.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor0_3.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor0_3.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor0_3.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor0_3.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor0_3.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor0_3.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor0_3.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

exclusive_monitor1_0.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor1_0.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor1_0.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor1_0.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor1_0.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor1_0.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor1_0.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor1_0.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor1_0.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

exclusive_monitor1_1.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor1_1.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor1_1.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor1_1.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor1_1.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor1_1.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor1_1.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor1_1.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor1_1.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

exclusive_monitor1_2.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor1_2.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor1_2.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor1_2.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor1_2.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor1_2.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor1_2.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor1_2.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor1_2.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

exclusive_monitor1_3.apply_access_width_criteria_to_non_excl_stores

Apply the given exclusive store width matching criteria to non-exclusive stores

Type

bool

Default value

0x1

exclusive_monitor1_3.clear_on_strex_address_mismatch

Whether monitor is cleared when strex fails due to address mismatch

Type

bool

Default value

0x1

exclusive_monitor1_3.enable_component

Enable component

Type

bool

Default value

0x1

exclusive_monitor1_3.exclusive_monitor_clear_on_atomic_from_same_master

Monitor atomics from the same master

Type

bool

Default value

0x1

exclusive_monitor1_3.match_secure_state

Treat the secure state like an address bit

Type

bool

Default value

0x1

exclusive_monitor1_3.monitor_access_level

0: Monitor all accesses, 1: Monitor all accesses except WriteBack, 2: Only monitor accesses with memory type NonCacheable or Device

Type

int

Default value

0x0

exclusive_monitor1_3.monitor_non_excl_stores

Monitor non-exclusive stores from the same master

Type

bool

Default value

0x0

exclusive_monitor1_3.number_of_monitors

Number of monitors

Type

int

Default value

0x8

exclusive_monitor1_3.shareability_domain

Maximum shareability domain of interest, transactions outside of the domain will pass through un-monitored (0-non-shared, 1-inner, 2-outer, 3-system)

Type

int

Default value

0x3

revision

Revision

Type

string

Default value

r1p2

3.10.78 PL350_SMC_NAND_FLASH

A NAND Flash implementation which works with PL350. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1147: IP revisions support

Revision	Quality level
r1p2	Full support
r2p2	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL350_SMC_NAND_FLASH

This model has the following Iris instances:

Table 3-1148: PL350_SMC_NAND_FLASH Iris instances

InstanceName	ComponentName
PL350_SMC_NAND_FLASH	PL350_SMC_NAND_FLASH
PL350_SMC_NAND_FLASH.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-1149: PL350_SMC_NAND_FLASH MTI instances

InstanceName	ComponentName
PL350_SMC_NAND_FLASH.busslave	PVBusSlave

PL350_SMC_NAND_FLASH contains the following CADI targets:

- [PL350_SMC_NAND_FLASH](#)

About PL350_SMC_NAND_FLASH

Program the component as you would the hardware.

Ports for PL350_SMC_NAND_FLASH

Table 3-1150: Ports

Name	Protocol	Type	Description
irq	Signal	Master	Interrupt signaling.
pvbus	PVBus	Slave	Slave port for connection to PV bus master/decoder.

Parameters for PL350_SMC_NAND_FLASH

DEVICE_1
Device manufacturer code
Type
int
Default value
0xec

DEVICE_2
Device code
Type
int
Default value
0xda

DEVICE_3
Device 3rd cycle code
Type
int
Default value
0x80

DEVICE_4
Device 4th cycle code
Type
int
Default value
0x15

DEVICE_NAME
Device Name
Type
string

Default value

Samsung K9F1G08U0M

NAND_BLOCK_COUNT

number of blocks in the flash device

Type

int

Default value

0x800

NAND_FLASH_SIZE

flash size in byte

Type

int

Default value

0x10800000

NAND_PAGE_COUNT_PER_BLOCK

number of pages in each block

Type

int

Default value

0x40

NAND_PAGE_SIZE

page size

Type

int

Default value

0x840

NAND_SPARE_SIZE_PER_PAGE

Spare size per page

Type

int

Default value

0x40

NAND_VALID_SIZE_PER_PAGE

valid page size

Type

int

Default value

0x800

3.10.79 PL370_HDLCD

ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1151: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL370_HDLCD

This model has the following Iris instances:

Table 3-1152: PL370_HDLCD Iris instances

InstanceName	ComponentName
PL370_HDLCD	PL370_HDLCD
PL370_HDLCD.busmaster	PVBusMaster
PL370_HDLCD.busslave	PVBusSlave
PL370_HDLCD.timer	ClockTimerThread
PL370_HDLCD.timer.timer	ClockTimerThread64
PL370_HDLCD.timer.timer.thread	SchedulerThread
PL370_HDLCD.timer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-1153: PL370_HDLCD MTI instances

InstanceName	ComponentName
PL370_HDLCD.busmaster	PVBusMaster
PL370_HDLCD.busslave	PVBusSlave

PL370_HDLCD contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- PL370_HDLCD

- SchedulerThread
- SchedulerThreadEvent



Too fast a pixel clock can slow the rest of the simulation.

Ports for PL370_HDLCD

Table 3-1154: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Master clock input, typically 24MHz, to drive pixel clock timing.
display	LCD	Master	Connection to visualization component.
intr	Signal	Master	Interrupt signaling line for flyback events.
pvbus	PVBus	Slave	Slave port for connection to PV bus master/decoder.
pvbus_m	PVBus	Master	DMA port for collecting video data from memory/framebuffer.

Parameters for PL370_HDLCD

diagnostics

Diagnostics level

Type

int

Default value

0x0

disable_snooping_dma

Disable DMA snooping

Type

bool

Default value

0x0

force_frame_rate

Force frame rate to the value of the parameter in frames per simulated second, regardless of the input clock. When 0, use the input clock as a pixel clock

Type

int

Default value

0x32

3.10.80 PL390_GIC

Generic Interrupt Controller (PL390). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1155: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PL390_GIC

This model has the following Iris instances:

Table 3-1156: PL390_GIC Iris instances

InstanceName	ComponentName
PL390_GIC	PL390_GIC
PL390_GIC.busslave_cpu	PVBusSlave
PL390_GIC.busslave_distributor	PVBusSlave

This model has the following MTI trace components:

Table 3-1157: PL390_GIC MTI instances

InstanceName	ComponentName
PL390_GIC.busslave_cpu	PVBusSlave
PL390_GIC.busslave_distributor	PVBusSlave

PL390_GIC contains the following CADI targets:

- PL390_GIC

About PL390_GIC

The GIC provides support for three interrupt types:

- Software Generated Interrupts (SGI)
- Private Peripheral Interrupts (PPI)
- Shared Peripheral Interrupts (SPI)

You can set:

- Security state for an interrupt
- Priority state for an interrupt
- Enabling or disabling state for an interrupt
- Processors that receive an interrupt

A processor interface consists of a pair of interfaces called `pvbus_cpu` and `pvbus_distributor`. The `enable_c<n>` and `match_c<n>` signals identify the originator of a transaction on `pvbus_cpu`. Similarly, `enable_d<n>` and `match_d<n>` signals identify the originator of a transaction on `pvbus_distributor`. `<n>` corresponds to the number of a processor interface.



Note

To reduce compile time, the registers are not available by default. To activate them, uncomment either of the following statements in `PL390_GIC.lisa`:

```
// #define FEW_CADI_REGISTER
// #define ALL_CADI_REGISTER
```

Ports for PL390_GIC

Table 3-1158: Ports

Name	Protocol	Type	Description
<code>cfgsdisable</code>	Signal	Slave	Set preventing write accesses to security-critical configuration registers.
<code>enable_c[8]</code>	ValueState	Slave	Compared with masked PVBUS master id to select processor interface: $(\text{master_id} \& \text{enable_c}\langle n \rangle) == \text{match_c}\langle n \rangle$.
<code>enable_d[8]</code>	ValueState	Slave	Compared with masked PVBUS master id to select distributor interface: $(\text{master_id} \& \text{enable_d}\langle n \rangle) == \text{match_d}\langle n \rangle$.
<code>legacy_nfiq[8]</code>	Signal	Slave	Legacy FIQ interrupt for processor Interface <code><n></code> .
<code>legacy_nirq[8]</code>	Signal	Slave	Input interrupt signals.
<code>match_c[8]</code>	ValueState	Slave	Mask on the PVBUS master id to select processor interface: $(\text{master_id} \& \text{enable_c}\langle n \rangle) == \text{match_c}\langle n \rangle$.
<code>match_d[8]</code>	ValueState	Slave	Mask on the PVBUS master id to select distributor interface: $(\text{master_id} \& \text{enable_d}\langle n \rangle) == \text{match_d}\langle n \rangle$.
<code>nfiq[8]</code>	Signal	Master	Send out FIQ signal to processor <code><n></code> .
<code>nirq[8]</code>	Signal	Master	Send out IRQ signal to processor <code><n></code> .
<code>ppi_c0[16]</code>	Signal	Slave	Private peripheral interrupt for processor 0 (<code>num_cpus</code> = 1).
<code>ppi_c1[16]</code>	Signal	Slave	Private peripheral interrupt for processor 1 (<code>num_cpus</code> = 2).
<code>ppi_c2[16]</code>	Signal	Slave	Private peripheral interrupt for processor 2 (<code>num_cpus</code> = 3).
<code>ppi_c3[16]</code>	Signal	Slave	Private peripheral interrupt for processor 3 (<code>num_cpus</code> = 4).
<code>ppi_c4[16]</code>	Signal	Slave	Private peripheral interrupt for processor 4 (<code>num_cpus</code> = 5).
<code>ppi_c5[16]</code>	Signal	Slave	Private peripheral interrupt for processor 5 (<code>num_cpus</code> = 6).
<code>ppi_c6[16]</code>	Signal	Slave	Private peripheral interrupt for processor 6 (<code>num_cpus</code> = 7).
<code>ppi_c7[16]</code>	Signal	Slave	Private peripheral interrupt for processor 7 (<code>num_cpus</code> = 8).
<code>pvbus_cpu</code>	PVBUS	Slave	Slave port for connection to processor interface.
<code>pvbus_distributor</code>	PVBUS	Slave	Slave port for connection to distributor interface.
<code>reset_in</code>	Signal	Slave	Reset signal.
<code>spi[988]</code>	Signal	Slave	Shared peripheral interrupt inputs.

Parameters for PL390_GIC

ARCHITECTURE_VERSION

set architecture version in periph_id register

Type

int

Default value

0x1

AXI_IF

set interface type in peripheral identification register 8

Type

bool

Default value

0x1

C_ID_WIDTH

width of the cpu interface master id

Type

int

Default value

0x20

D_ID_WIDTH

width of the distributor interface master id

Type

int

Default value

0x20

ENABLE_LEGACY_FIQ

provide legacy fiq interrupt inputs

Type

bool

Default value

0x1

ENABLE_LEGACY_IRQ

provide legacy irq interrupt inputs

Type

bool

Default value

0x1

ENABLE_PPI_EDGE

ppi edge sensitive

Type

bool

Default value

0x0

ENABLE_TRUSTZONE

support trustzone

Type

bool

Default value

0x1

INIT_ENABLE_C0

initial value of register ENABLE_C0

Type

int

Default value

0xffffffff

INIT_ENABLE_C1

initial value of register ENABLE_C1

Type

int

Default value

0xffffffff

INIT_ENABLE_C2

initial value of register ENABLE_C2

Type

int

Default value

0xffffffff

INIT_ENABLE_C3

initial value of register ENABLE_C3

Type

int

Default value

0xffffffff

INIT_ENABLE_C4

initial value of register ENABLE_C4

Type

int

Default value

0xffffffff

INIT_ENABLE_C5

initial value of register ENABLE_C5

Type

int

Default value

0xffffffff

INIT_ENABLE_C6

initial value of register ENABLE_C6

Type

int

Default value

0xffffffff

INIT_ENABLE_C7

initial value of register ENABLE_C7

Type

int

Default value

0xffffffff

INIT_ENABLE_D0

initial value of register ENABLE_D0

Type

int

Default value

0xffffffff

INIT_ENABLE_D1

initial value of register ENABLE_D1

Type

int

Default value

0xffffffff

INIT_ENABLE_D2

initial value of register ENABLE_D2

Type

int

Default value

0xffffffff

INIT_ENABLE_D3

initial value of register ENABLE_D3

Type

int

Default value

0xffffffff

INIT_ENABLE_D4

initial value of register ENABLE_D4

Type

int

Default value

0xffffffff

INIT_ENABLE_D5

initial value of register ENABLE_D5

Type

int

Default value

0xffffffff

INIT_ENABLE_D6

initial value of register ENABLE_D6

Type

int

Default value

0xffffffff

INIT_ENABLE_D7

initial value of register ENABLE_D7

Type

int

Default value

0xffffffff

INIT_MATCH_C0

initial value of register MATCH_C0

Type

int

Default value

0x0

INIT_MATCH_C1

initial value of register MATCH_C1

Type

int

Default value

0x1

INIT_MATCH_C2

initial value of register MATCH_C2

Type

int

Default value

0x2

INIT_MATCH_C3

initial value of register MATCH_C3

Type

int

Default value

0x3

INIT_MATCH_C4

initial value of register MATCH_C4

Type

int

Default value

0x4

INIT_MATCH_C5

initial value of register MATCH_C5

Type

int

Default value

0x5

INIT_MATCH_C6

initial value of register MATCH_C6

Type

int

Default value

0x6

INIT_MATCH_C7

initial value of register MATCH_C7

Type

int

Default value

0x7

INIT_MATCH_D0

initial value of register MATCH_D0

Type

int

Default value

0x0

INIT_MATCH_D1

initial value of register MATCH_D1

Type

int

Default value

0x1

INIT_MATCH_D2

initial value of register MATCH_D2

Type

int

Default value

0x2

INIT_MATCH_D3

initial value of register MATCH_D3

Type

int

Default value

0x3

INIT_MATCH_D4

initial value of register MATCH_D4

Type

int

Default value

0x4

INIT_MATCH_D5

initial value of register MATCH_D5

Type

int

Default value

0x5

INIT_MATCH_D6

initial value of register MATCH_D6

Type

int

Default value

0x6

INIT_MATCH_D7

initial value of register MATCH_D7

Type

int

Default value

0x7

NUM_CPU

number of cpu interfaces

Type

int

Default value

0x8

NUM_LSPI

number of lockable shared peripheral interrupts

Type

int

Default value

0x1f

NUM_PPI

number of peripheral interrupts

Type

int

Default value

0x10

NUM_PRIORITY_LEVELS

number of priority levels

Type

int

Default value

0x100

NUM_SGI

number of software generated interrupts

Type

int

Default value

0x10

NUM_SPI

number of shared peripheral interrupts

Type

int

Default value

0x3dc

3.10.81 PPUMTWakerequest

Power Policy Unit (PPU) v8.2 Multi-threaded Core Wakerequest Logic. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1159: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Ports for PPUMTWakerequest

Table 3-1160: Ports

Name	Protocol	Type	Description
cpu_pchannel_m	PChannel	Master	-
ppu_pchannel_s	PChannel	Slave	-
thread_wake_request[2]	Signal	Slave	-
wakerequest	Signal	Master	-

Parameters for PPUMTWakerequest

mt_mode

Multi-threaded mode

Type

bool

Default value

0x0

thread0_op_mode_bit

Thread0 Operation Mode bit of DEVPACTIVE

Type

int

Default value

0x10

thread1_op_mode_bit

Thread1 Operation Mode bit of DEVPACTIVE

Type

int

Default value

0x11

3.10.82 PPUv0

Power Policy Unit (PPU) v0.8 architectural model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1161: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PPUv0

This model has the following Iris instances:

Table 3-1162: PPUv0 Iris instances

InstanceName	ComponentName
PPUv0	PPUv0
PPUv0.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-1163: PPUv0 MTI instances

InstanceName	ComponentName
PPUv0	PPUv0
PPUv0.busslave	PVBusSlave

PPUv0 contains the following CADI targets:

- [PPUv0](#)

Ports for PPUv0

Table 3-1164: Ports

Name	Protocol	Type	Description
irq	Signal	Master	-
powerdown	Signal	Master	-
ppuhwstat	Value	Master	-
pvbus_s	PVBus	Slave	-
smpen	Signal	Slave	-
standbywfi	Signal	Slave	-
wakerequest	Signal	Slave	-

Parameters for PPUv0

default_power_state_on

Default power state ON

Type

bool

Default value

0x0

device_channels

Number of device channels (0: P-Channel, 1-8: Q-Channels)

Type

int

Default value

0x0

dynamic_off

Dynamic Off

Type

bool

Default value

0x0

dynamic_on

Dynamic On

Type

bool

Default value

0x0

dynamic_warm_reset

Dynamic Warm Reset

Type

bool

Default value

0x0

full_ret

Full Retention (0: not supported, 1: static, 2: dynamic)

Type

int

Default value

0x0

func_ret

Functional Retention (0: not supported, 1: static, 2: dynamic)

Type

int

Default value

0x0

logic_ret

Logic Retention (0: not supported, 1: static, 2: dynamic)

Type

int

Default value

0x0

mem_off

Memory Off (0: not supported, 1: static, 2: dynamic)

Type

int

Default value

0x0

mem_ret

Memory Retention (0: not supported, 1: static, 2: dynamic)

Type

int

Default value

0x0

revision

Revision

Type

string

Default value

r0p0

use_active_signal

Use device-active signal

Type

bool

Default value

0x0

3.10.83 PPUv1

Power Policy Unit (PPU) v1.1 architectural model. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1165: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for PPUv1

This model has the following Iris instances:

Table 3-1166: PPUv1 Iris instances

InstanceName	ComponentName
PPUv1	PPUv1
PPUv1.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-1167: PPUv1 MTI instances

InstanceName	ComponentName
PPUv1	PPUv1
PPUv1.busslave	PVBusSlave

PPUv1 contains the following CADI targets:

- PPUv1

About PPUv1

Software can determine which features the PPU supports by reading the PPU Identification Register 0, `PPU_IDR0` and the PPU Identification Register 1, `PPU_IDR1`.

The following power policies are offered by the PPU model, in order of increasing priority:

- Off.
- Emulated Off.
- Memory Retention.
- Emulated Memory Retention.
- Logic Retention.
- Full Retention.
- Memory Off.
- Functional Retention.
- On.
- Warm Reset.
- Debug Recovery Reset.

For the power mode transition rules, see [Arm Power Policy Unit Architecture Specification](#).

There are 16 operating mode values. The meaning of these values is specific to the device that is connected to the PPU. The operating mode can only be configured to change during a power transition of ON to ON.

The PPU model supports static and dynamic transitions on the P-Channel interface. It does not yet support Q-Channel.

`DEVPACTIVE` and `DEVPSTATE` have the following bit encodings:

`DEVPACTIVE` bits [10:0]

Each bit indicates a required power mode.

`DEVPSTATE` bits [3:0]

The integer formed by this bitfield indicates a power mode.

`DEVPACTIVE` bits [23:16]

Operating mode. The interpretation of these bits depends on the `DEVPACTIVE` use model (Ladder or Independent).

`DEVPSTATE` bits [7:4]

The integer formed by this bitfield indicates an operating mode.

Communication over the Low Power Interface (`PREQUEST` and `PACTIVE`) uses blocking calls and does not model any delays. See [2.5.1 PChannel protocol](#) on page 89 for further details.

For the AMBA Low Power Interface Specification Arm Q-Channel and P-Channel Interfaces, see [AMBA Low Power Interface Specification](#).

For static transitions, software sets the policy as the required power mode. The PPU then sends a `PREQUEST` with the required power state to the attached device. The device can `ACCEPT` or `DENY` it. For dynamic transitions, software sets the policy as a minimum power mode. Based on whether the device has sent a signal using `DEVPACTIVE`, the PPU sends a `PREQUEST` with the required power state to the attached device. The device can `ACCEPT` or `DENY` it.

The PPU model is automatically reset by the simulation engine when the model starts up. Reset can also occur through the `reset_in` port. The PPU model is reset only when the signal value is `signal::Set`. Use `signal::Set` instead of zero, its integer value, to prevent unexpected behavior.

The `ppuhwstat` port notifies the power state change inside the PPU and the definition of each bit is the same as `DEVPACTIVE[10:0]`.

The `smpen` and `standbywfi` ports are defined in PPUv0 and are not supported in PPUv1.

Differences between the model and the RTL

- Q-Channel is not supported
- The PPU model has been validated with devices supporting only ON and OFF power modes. Arm has not tested the case where a connected device supports other power modes offered by the PPU.

Ports for PPUv1

Table 3-1168: Ports

Name	Protocol	Type	Description
dev_clk_en_out	Signal	Master	Domain clock enable
dev_emu_clk_en_out	Signal	Master	Domain emulated mode clock enable
dev_emu_isolaten_out	Signal	Master	Domain emulated isolation control.
dev_isolaten_out	Signal	Master	Domain isolation control.
dev_poresetn_out	Signal	Master	Domain power on reset
dev_ret_reseten_out	Signal	Master	Domain retention reset.
dev_warm_reseten_out	Signal	Master	Domain warm reset
devpactive	PChannel	Master	P-Channel port
irq	Signal	Master	PPU IRQ signal
powerdown	Signal	Master	Notify whether or not the PPU is in OFF state.
ppuhwstat	Value	Master	Notify the power state change inside the PPU. The definition of each bit is the same as DEVPACTIVE[10:0].
pvbus_s	PVBus	Slave	PPU APB bus slave port
reset_in	Signal	Slave	PPU reset signal input
wakerequest	Signal	Slave	Input port for the wakerequest signal. It is ORed with PACTIVE[8] (ON) inside the PPU as input to PPU DEVPACTIVE[8] (ON). The "is_core_ppu" parameter controls whether there is additional logic to hold this signal until the PPU is in OFF/OFF_EMU state.

Parameters for PPUv1

RevD_support

Whether to support Rev D locked IRQ

Type

bool

Default value

0x1

dbg_recov

Debug Recovery Reset (0: not supported, 1: static mode only, 2: both dynamic & static mode)

Type

int

Default value

0x0

default_op_dyn_en

Whether to enable operating mode dynamic transition by default

Type

bool

Default value

0x0

default_op_policy

Default operating policy

Type

int

Default value

0x0

default_power_state_on

Default power state ON

Type

bool

Default value

0x0

default_pwr_dyn_en

Whether to enable dynamic power mode transition by default

Type

bool

Default value

0x0

device_channels

Number of device channels (0: P-Channel, 1-8: Q-Channels)

Type

int

Default value

0x0

dynamic_off

Dynamic Off

Type

bool

Default value

0x0

dynamic_on

Dynamic On

Type

bool

Default value

0x0

dynamic_warm_reset

Dynamic Warm Reset

Type

bool

Default value

0x0

full_ret

Full Retention (0: not supported, 1: static mode only, 2: both dynamic & static mode)

Type

int

Default value

0x0

func_ret

Functional Retention (0: not supported, 1: static mode only, 2: both dynamic & static mode)

Type

int

Default value

0x0

is_core_ppu

PPU is core_ppu type which means wake_request would wait till PPU is OFF/OFF_EMU

Type

bool

Default value

0x0

lock_support

Whether to support OFF lock feature

Type

bool

Default value

0x1

logic_ret

Logic Retention (0: not supported, 1: static, 2: dynamic)

Type

int

Default value

0x0

mem_off

Memory Off (0: not supported, 1: static mode only, 2: both dynamic & static mode)

Type

int

Default value

0x0

mem_ret

Memory Retention (0: not supported, 1: static mode only, 2: both dynamic & static mode)

Type

int

Default value

0x0

mem_ret_emu

Emulated Memory Retention (0: not supported, 1: static mode only, 2: both dynamic & static mode)

Type

int

Default value

0x0

num_opmode_cfg

Number of operating modes

Type

int

Default value

0x0

off_emu

Emulated Off (0: not supported, 1: static mode only, 2: both dynamic & static mode)

Type

int

Default value

0x0

off_mem_ret_trans_cfg

OFF to MEM_RET direct transition configuration (0: not allowed, 1: allowed)

Type

bool

Default value

0x0

op_active_cfg

Operating mode active configuration (0: Ladder use model, 1: Independent user model)

Type

int

Default value

0x0

revision

Revision

Type

string

Default value

r1p1

use_active_signal

Use device-active signal

Type

bool

Default value

0x0

3.10.84 PPUv1_Cluster_Wakerequest_Logic

PPUv1 wake request stall logic. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1169: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Ports for PPUv1_Cluster_Wakerequest_Logic

Table 3-1170: Ports

Name	Protocol	Type	Description
cluster_wake_request	Signal	Master	-
core_wake_request_in[8]	Signal	Slave	-
core_wake_request_out[8]	Signal	Master	-
ppuhwstat	Value	Slave	-
reset_in	Signal	Slave	-

Parameters for PPUv1_Cluster_Wakerequest_Logic

core_ppu_wakerequest_stall_condition_after_reset

Set Stall Condition of Core WakeRequest (from GIC) for Core PPU after reset

Type

bool

Default value

0x0

disable_core_ppu_wakerequest_input_stall

Disable wakerequest input stall of Core PPU. This feature is enabled by default to mimic the P-Channel request stall when Cluster PPU is in OFF.

Type

bool

Default value

0x0

enable_cluster_wakeup_if_cluster_on_funcret

enable cluster wakeup logic. If it's disabled, core_wake_request_in[x] will be directly connected to core_wake_request_out[x] and cluster_wake_request port is disabled

Type

bool

Default value

0x1

3.10.85 RSS_CPU_Private_Region

RSS CPU processor private region. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1171: IP revisions support

Revision	Quality level
1.46	Alpha support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for RSS_CPU_Private_Region

This model has the following Iris instances:

Table 3-1172: RSS_CPU_Private_Region Iris instances

InstanceName	ComponentName
RSS_CPU_Private_Region	RSS_CPU_Private_Region
RSS_CPU_Private_Region.apb_nonsecure	PVBUSSlave
RSS_CPU_Private_Region.apb_secure	PVBUSSlave

This model has the following MTI trace components:

Table 3-1173: RSS_CPU_Private_Region MTI instances

InstanceName	ComponentName
RSS_CPU_Private_Region.apb_nonsecure	PVBUSSlave
RSS_CPU_Private_Region.apb_secure	PVBUSSlave

RSS_CPU_Private_Region contains the following CADI targets:

- [RSS_CPU_Private_Region](#)

Ports for RSS_CPU_Private_Region

Table 3-1174: Ports

Name	Protocol	Type	Description
apb_nonsecure	PVBUS	Slave	-
apb_secure	PVBUS	Slave	secure & non-secure Subordinate APB Interface
reset_in	Signal	Slave	Reset in signal

Parameters for RSS_CPU_Private_Region

CPUID_RESET_VALUE

CPUID Registers Reset Value

Type

int

Default value

0x0

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type

int

Default value

0x2

3.10.86 SMMUv3AEM

This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1175: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for SMMUv3AEM

This model has the following Iris instances:

Table 3-1176: SMMUv3AEM Iris instances

InstanceName	ComponentName
SMMUv3AEM	SMMUv3AEM
SMMUv3AEM.register_file[0]	PVBusSlave
SMMUv3AEM.service_request_tbu[0]	PVBusSlave
SMMUv3AEM.service_request_tbu[10]	PVBusSlave
SMMUv3AEM.service_request_tbu[11]	PVBusSlave
SMMUv3AEM.service_request_tbu[12]	PVBusSlave
SMMUv3AEM.service_request_tbu[13]	PVBusSlave
SMMUv3AEM.service_request_tbu[14]	PVBusSlave
SMMUv3AEM.service_request_tbu[15]	PVBusSlave
SMMUv3AEM.service_request_tbu[16]	PVBusSlave
SMMUv3AEM.service_request_tbu[17]	PVBusSlave
SMMUv3AEM.service_request_tbu[18]	PVBusSlave
SMMUv3AEM.service_request_tbu[19]	PVBusSlave
SMMUv3AEM.service_request_tbu[1]	PVBusSlave
SMMUv3AEM.service_request_tbu[20]	PVBusSlave
SMMUv3AEM.service_request_tbu[21]	PVBusSlave
SMMUv3AEM.service_request_tbu[22]	PVBusSlave
SMMUv3AEM.service_request_tbu[23]	PVBusSlave
SMMUv3AEM.service_request_tbu[24]	PVBusSlave
SMMUv3AEM.service_request_tbu[25]	PVBusSlave

InstanceName	ComponentName
SMMUv3AEM.service_request_tbu[26]	PVBusSlave
SMMUv3AEM.service_request_tbu[27]	PVBusSlave
SMMUv3AEM.service_request_tbu[28]	PVBusSlave
SMMUv3AEM.service_request_tbu[29]	PVBusSlave
SMMUv3AEM.service_request_tbu[2]	PVBusSlave
SMMUv3AEM.service_request_tbu[30]	PVBusSlave
SMMUv3AEM.service_request_tbu[31]	PVBusSlave
SMMUv3AEM.service_request_tbu[32]	PVBusSlave
SMMUv3AEM.service_request_tbu[33]	PVBusSlave
SMMUv3AEM.service_request_tbu[34]	PVBusSlave
SMMUv3AEM.service_request_tbu[35]	PVBusSlave
SMMUv3AEM.service_request_tbu[36]	PVBusSlave
SMMUv3AEM.service_request_tbu[37]	PVBusSlave
SMMUv3AEM.service_request_tbu[38]	PVBusSlave
SMMUv3AEM.service_request_tbu[39]	PVBusSlave
SMMUv3AEM.service_request_tbu[3]	PVBusSlave
SMMUv3AEM.service_request_tbu[40]	PVBusSlave
SMMUv3AEM.service_request_tbu[41]	PVBusSlave
SMMUv3AEM.service_request_tbu[42]	PVBusSlave
SMMUv3AEM.service_request_tbu[43]	PVBusSlave
SMMUv3AEM.service_request_tbu[44]	PVBusSlave
SMMUv3AEM.service_request_tbu[45]	PVBusSlave
SMMUv3AEM.service_request_tbu[46]	PVBusSlave
SMMUv3AEM.service_request_tbu[47]	PVBusSlave
SMMUv3AEM.service_request_tbu[48]	PVBusSlave
SMMUv3AEM.service_request_tbu[49]	PVBusSlave
SMMUv3AEM.service_request_tbu[4]	PVBusSlave
SMMUv3AEM.service_request_tbu[50]	PVBusSlave
SMMUv3AEM.service_request_tbu[51]	PVBusSlave
SMMUv3AEM.service_request_tbu[52]	PVBusSlave
SMMUv3AEM.service_request_tbu[53]	PVBusSlave
SMMUv3AEM.service_request_tbu[54]	PVBusSlave
SMMUv3AEM.service_request_tbu[55]	PVBusSlave
SMMUv3AEM.service_request_tbu[56]	PVBusSlave
SMMUv3AEM.service_request_tbu[57]	PVBusSlave
SMMUv3AEM.service_request_tbu[58]	PVBusSlave
SMMUv3AEM.service_request_tbu[59]	PVBusSlave
SMMUv3AEM.service_request_tbu[5]	PVBusSlave
SMMUv3AEM.service_request_tbu[60]	PVBusSlave

InstanceName	ComponentName
SMMUv3AEM.service_request_tbu[61]	PVBusSlave
SMMUv3AEM.service_request_tbu[62]	PVBusSlave
SMMUv3AEM.service_request_tbu[63]	PVBusSlave
SMMUv3AEM.service_request_tbu[6]	PVBusSlave
SMMUv3AEM.service_request_tbu[7]	PVBusSlave
SMMUv3AEM.service_request_tbu[8]	PVBusSlave
SMMUv3AEM.service_request_tbu[9]	PVBusSlave
SMMUv3AEM.tbu[0]	PVBusMapper
SMMUv3AEM.tbu[10]	PVBusMapper
SMMUv3AEM.tbu[11]	PVBusMapper
SMMUv3AEM.tbu[12]	PVBusMapper
SMMUv3AEM.tbu[13]	PVBusMapper
SMMUv3AEM.tbu[14]	PVBusMapper
SMMUv3AEM.tbu[15]	PVBusMapper
SMMUv3AEM.tbu[16]	PVBusMapper
SMMUv3AEM.tbu[17]	PVBusMapper
SMMUv3AEM.tbu[18]	PVBusMapper
SMMUv3AEM.tbu[19]	PVBusMapper
SMMUv3AEM.tbu[1]	PVBusMapper
SMMUv3AEM.tbu[20]	PVBusMapper
SMMUv3AEM.tbu[21]	PVBusMapper
SMMUv3AEM.tbu[22]	PVBusMapper
SMMUv3AEM.tbu[23]	PVBusMapper
SMMUv3AEM.tbu[24]	PVBusMapper
SMMUv3AEM.tbu[25]	PVBusMapper
SMMUv3AEM.tbu[26]	PVBusMapper
SMMUv3AEM.tbu[27]	PVBusMapper
SMMUv3AEM.tbu[28]	PVBusMapper
SMMUv3AEM.tbu[29]	PVBusMapper
SMMUv3AEM.tbu[2]	PVBusMapper
SMMUv3AEM.tbu[30]	PVBusMapper
SMMUv3AEM.tbu[31]	PVBusMapper
SMMUv3AEM.tbu[32]	PVBusMapper
SMMUv3AEM.tbu[33]	PVBusMapper
SMMUv3AEM.tbu[34]	PVBusMapper
SMMUv3AEM.tbu[35]	PVBusMapper
SMMUv3AEM.tbu[36]	PVBusMapper
SMMUv3AEM.tbu[37]	PVBusMapper
SMMUv3AEM.tbu[38]	PVBusMapper

InstanceName	ComponentName
SMMUv3AEM.tbu[39]	PVBusMapper
SMMUv3AEM.tbu[3]	PVBusMapper
SMMUv3AEM.tbu[40]	PVBusMapper
SMMUv3AEM.tbu[41]	PVBusMapper
SMMUv3AEM.tbu[42]	PVBusMapper
SMMUv3AEM.tbu[43]	PVBusMapper
SMMUv3AEM.tbu[44]	PVBusMapper
SMMUv3AEM.tbu[45]	PVBusMapper
SMMUv3AEM.tbu[46]	PVBusMapper
SMMUv3AEM.tbu[47]	PVBusMapper
SMMUv3AEM.tbu[48]	PVBusMapper
SMMUv3AEM.tbu[49]	PVBusMapper
SMMUv3AEM.tbu[4]	PVBusMapper
SMMUv3AEM.tbu[50]	PVBusMapper
SMMUv3AEM.tbu[51]	PVBusMapper
SMMUv3AEM.tbu[52]	PVBusMapper
SMMUv3AEM.tbu[53]	PVBusMapper
SMMUv3AEM.tbu[54]	PVBusMapper
SMMUv3AEM.tbu[55]	PVBusMapper
SMMUv3AEM.tbu[56]	PVBusMapper
SMMUv3AEM.tbu[57]	PVBusMapper
SMMUv3AEM.tbu[58]	PVBusMapper
SMMUv3AEM.tbu[59]	PVBusMapper
SMMUv3AEM.tbu[5]	PVBusMapper
SMMUv3AEM.tbu[60]	PVBusMapper
SMMUv3AEM.tbu[61]	PVBusMapper
SMMUv3AEM.tbu[62]	PVBusMapper
SMMUv3AEM.tbu[63]	PVBusMapper
SMMUv3AEM.tbu[6]	PVBusMapper
SMMUv3AEM.tbu[7]	PVBusMapper
SMMUv3AEM.tbu[8]	PVBusMapper
SMMUv3AEM.tbu[9]	PVBusMapper

This model has the following MTI trace components:

Table 3-1177: SMMUv3AEM MTI instances

InstanceName	ComponentName
SMMUv3AEM	SMMUv3AEM
SMMUv3AEM.register_file[0]	PVBusSlave
SMMUv3AEM.service_request_tbu[0]	PVBusSlave

InstanceName	ComponentName
SMMUv3AEM.service_request_tbu[10]	PVBusSlave
SMMUv3AEM.service_request_tbu[11]	PVBusSlave
SMMUv3AEM.service_request_tbu[12]	PVBusSlave
SMMUv3AEM.service_request_tbu[13]	PVBusSlave
SMMUv3AEM.service_request_tbu[14]	PVBusSlave
SMMUv3AEM.service_request_tbu[15]	PVBusSlave
SMMUv3AEM.service_request_tbu[16]	PVBusSlave
SMMUv3AEM.service_request_tbu[17]	PVBusSlave
SMMUv3AEM.service_request_tbu[18]	PVBusSlave
SMMUv3AEM.service_request_tbu[19]	PVBusSlave
SMMUv3AEM.service_request_tbu[1]	PVBusSlave
SMMUv3AEM.service_request_tbu[20]	PVBusSlave
SMMUv3AEM.service_request_tbu[21]	PVBusSlave
SMMUv3AEM.service_request_tbu[22]	PVBusSlave
SMMUv3AEM.service_request_tbu[23]	PVBusSlave
SMMUv3AEM.service_request_tbu[24]	PVBusSlave
SMMUv3AEM.service_request_tbu[25]	PVBusSlave
SMMUv3AEM.service_request_tbu[26]	PVBusSlave
SMMUv3AEM.service_request_tbu[27]	PVBusSlave
SMMUv3AEM.service_request_tbu[28]	PVBusSlave
SMMUv3AEM.service_request_tbu[29]	PVBusSlave
SMMUv3AEM.service_request_tbu[2]	PVBusSlave
SMMUv3AEM.service_request_tbu[30]	PVBusSlave
SMMUv3AEM.service_request_tbu[31]	PVBusSlave
SMMUv3AEM.service_request_tbu[32]	PVBusSlave
SMMUv3AEM.service_request_tbu[33]	PVBusSlave
SMMUv3AEM.service_request_tbu[34]	PVBusSlave
SMMUv3AEM.service_request_tbu[35]	PVBusSlave
SMMUv3AEM.service_request_tbu[36]	PVBusSlave
SMMUv3AEM.service_request_tbu[37]	PVBusSlave
SMMUv3AEM.service_request_tbu[38]	PVBusSlave
SMMUv3AEM.service_request_tbu[39]	PVBusSlave
SMMUv3AEM.service_request_tbu[3]	PVBusSlave
SMMUv3AEM.service_request_tbu[40]	PVBusSlave
SMMUv3AEM.service_request_tbu[41]	PVBusSlave
SMMUv3AEM.service_request_tbu[42]	PVBusSlave
SMMUv3AEM.service_request_tbu[43]	PVBusSlave
SMMUv3AEM.service_request_tbu[44]	PVBusSlave
SMMUv3AEM.service_request_tbu[45]	PVBusSlave

InstanceName	ComponentName
SMMUv3AEM.service_request_tbu[46]	PVBusSlave
SMMUv3AEM.service_request_tbu[47]	PVBusSlave
SMMUv3AEM.service_request_tbu[48]	PVBusSlave
SMMUv3AEM.service_request_tbu[49]	PVBusSlave
SMMUv3AEM.service_request_tbu[4]	PVBusSlave
SMMUv3AEM.service_request_tbu[50]	PVBusSlave
SMMUv3AEM.service_request_tbu[51]	PVBusSlave
SMMUv3AEM.service_request_tbu[52]	PVBusSlave
SMMUv3AEM.service_request_tbu[53]	PVBusSlave
SMMUv3AEM.service_request_tbu[54]	PVBusSlave
SMMUv3AEM.service_request_tbu[55]	PVBusSlave
SMMUv3AEM.service_request_tbu[56]	PVBusSlave
SMMUv3AEM.service_request_tbu[57]	PVBusSlave
SMMUv3AEM.service_request_tbu[58]	PVBusSlave
SMMUv3AEM.service_request_tbu[59]	PVBusSlave
SMMUv3AEM.service_request_tbu[5]	PVBusSlave
SMMUv3AEM.service_request_tbu[60]	PVBusSlave
SMMUv3AEM.service_request_tbu[61]	PVBusSlave
SMMUv3AEM.service_request_tbu[62]	PVBusSlave
SMMUv3AEM.service_request_tbu[63]	PVBusSlave
SMMUv3AEM.service_request_tbu[6]	PVBusSlave
SMMUv3AEM.service_request_tbu[7]	PVBusSlave
SMMUv3AEM.service_request_tbu[8]	PVBusSlave
SMMUv3AEM.service_request_tbu[9]	PVBusSlave
SMMUv3AEM.tb[0]	PVBusMapper
SMMUv3AEM.tb[10]	PVBusMapper
SMMUv3AEM.tb[11]	PVBusMapper
SMMUv3AEM.tb[12]	PVBusMapper
SMMUv3AEM.tb[13]	PVBusMapper
SMMUv3AEM.tb[14]	PVBusMapper
SMMUv3AEM.tb[15]	PVBusMapper
SMMUv3AEM.tb[16]	PVBusMapper
SMMUv3AEM.tb[17]	PVBusMapper
SMMUv3AEM.tb[18]	PVBusMapper
SMMUv3AEM.tb[19]	PVBusMapper
SMMUv3AEM.tb[1]	PVBusMapper
SMMUv3AEM.tb[20]	PVBusMapper
SMMUv3AEM.tb[21]	PVBusMapper
SMMUv3AEM.tb[22]	PVBusMapper

InstanceName	ComponentName
SMMUv3AEM.tbu [23]	PVBusMapper
SMMUv3AEM.tbu [24]	PVBusMapper
SMMUv3AEM.tbu [25]	PVBusMapper
SMMUv3AEM.tbu [26]	PVBusMapper
SMMUv3AEM.tbu [27]	PVBusMapper
SMMUv3AEM.tbu [28]	PVBusMapper
SMMUv3AEM.tbu [29]	PVBusMapper
SMMUv3AEM.tbu [2]	PVBusMapper
SMMUv3AEM.tbu [30]	PVBusMapper
SMMUv3AEM.tbu [31]	PVBusMapper
SMMUv3AEM.tbu [32]	PVBusMapper
SMMUv3AEM.tbu [33]	PVBusMapper
SMMUv3AEM.tbu [34]	PVBusMapper
SMMUv3AEM.tbu [35]	PVBusMapper
SMMUv3AEM.tbu [36]	PVBusMapper
SMMUv3AEM.tbu [37]	PVBusMapper
SMMUv3AEM.tbu [38]	PVBusMapper
SMMUv3AEM.tbu [39]	PVBusMapper
SMMUv3AEM.tbu [3]	PVBusMapper
SMMUv3AEM.tbu [40]	PVBusMapper
SMMUv3AEM.tbu [41]	PVBusMapper
SMMUv3AEM.tbu [42]	PVBusMapper
SMMUv3AEM.tbu [43]	PVBusMapper
SMMUv3AEM.tbu [44]	PVBusMapper
SMMUv3AEM.tbu [45]	PVBusMapper
SMMUv3AEM.tbu [46]	PVBusMapper
SMMUv3AEM.tbu [47]	PVBusMapper
SMMUv3AEM.tbu [48]	PVBusMapper
SMMUv3AEM.tbu [49]	PVBusMapper
SMMUv3AEM.tbu [4]	PVBusMapper
SMMUv3AEM.tbu [50]	PVBusMapper
SMMUv3AEM.tbu [51]	PVBusMapper
SMMUv3AEM.tbu [52]	PVBusMapper
SMMUv3AEM.tbu [53]	PVBusMapper
SMMUv3AEM.tbu [54]	PVBusMapper
SMMUv3AEM.tbu [55]	PVBusMapper
SMMUv3AEM.tbu [56]	PVBusMapper
SMMUv3AEM.tbu [57]	PVBusMapper
SMMUv3AEM.tbu [58]	PVBusMapper

InstanceName	ComponentName
SMMUv3AEM.tbv [59]	PVBusMapper
SMMUv3AEM.tbv [5]	PVBusMapper
SMMUv3AEM.tbv [60]	PVBusMapper
SMMUv3AEM.tbv [61]	PVBusMapper
SMMUv3AEM.tbv [62]	PVBusMapper
SMMUv3AEM.tbv [63]	PVBusMapper
SMMUv3AEM.tbv [6]	PVBusMapper
SMMUv3AEM.tbv [7]	PVBusMapper
SMMUv3AEM.tbv [8]	PVBusMapper
SMMUv3AEM.tbv [9]	PVBusMapper

SMMUv3AEM contains the following CADI targets:

- SMMUv3AEM

About SMMUv3AEM

The SMMUv3 Architecture Envelope Model component is an architectural model that implements the SMMUv3.0, SMMUv3.1, and SMMUv3.2 architectures for I/O virtualization of devices, except for the limitations listed below.

The SMMUv3 specifies that input addresses are conceptually 64 bits. The SMMUv3AEM model assumes that the input address is 64 bits. If the SoC has less than 64 bits as an input address bus then if the SoC wants to use the high address space (and use TT1) then it must sign extend the address from the upstream peripherals to get to 64 bits.

The SMMUv3AEM model deals with groups of transactions with the same attributes and a similar range of addresses. The mapping used is remembered by the bus infrastructure and is used for subsequent sufficiently similar transactions without requiring intervention from the SMMU model, and so will not be traced, for instance. The range of addresses that the mapping is valid for is determined by the SMMU model, depending on architectural and model-implementation details. However, as it is unaware of any sign-extension then the mapping that the SoC does, the SoC is responsible for subsequently narrowing the range of addresses for which this mapping is valid. Typically, this is done automatically when using `PVBusMapper`.

The model has the following limitations:

- It does not support:
 - RAS.
 - Power control.
 - AMBA® stash operations and destructive read operations are not supported on `PVBus` and also are not supported by the device.
 - PCIe-NoSnoop transactions.

- The PMU has limited functionality. Only a subset of the architecturally mandatory events are supported, as indicated by the `SMMU_PMC_G_CBEID0` fields. The PMU is intended for demonstration purposes only and for driver development.

Notes

- A bad configuration renders the model inactive.
- Some configurations can be adjusted by configuration pins. These are only sampled at the negative edge of the reset pin. If you want to use these pins then you must drive them before sending a negative edge on the reset pin. During simulation_reset the component driving them must also drive this transition again.
- Debug reads to the registers do not disturb the state.
- Writes to registers with Update flags, including debug writes, are ignored if the Update flag is already set to one.
- Debug and real accesses to the registers must be 32 bits or 64 bits.
- MSIs are issued using attributes determined by the parameter `msi_attribute_transform`, whilst Event queue writes are always issued with `ExtendedID=0`, `UserFlags=0`, `MasterID=0xFFFFFFFF`. Thus, if your system needs to distinguish MSI writes from Event queue writes, it can do so using this mechanism.
- If your system does table walks and queue accesses through TBUO (`separate_tw_msi_qs_port == false`), then care must be taken to distinguish table walk and queue traffic (with `MasterID=0xFFFFFFFF`) from normal translated traffic.
- If `SMMU_IDR1.TABLES_PRESET` or `SMMU_IDR1.QUEUES_PRESET` is set then see parameter `PRESET_REL_base_address` and the parameters it mentions. Embedded implementations of the SMMU are allowed to have the queues/stream table in a 'close' RAM, either on-chip or in the SMMU itself. For the model, it is up to the integrator to supply this memory and for the SMMU model to be able to access. Thus if the actual hardware has the memory built into the SMMU then it will be necessary for the integrator to wrap this model with a bus decoder and a memory model to more closely model the embedded implementation.

Security State Determination (SSD)

The model uses the term SSD to mean the security state that the transaction, register, structure, or event belongs to. In the SMMUv3 architecture, the sideband signal `SEC_SID` holds this information for the transactions, but uses a different encoding.

Before RME, `SEC_SID` was a one-bit signal. With RME, in the hardware, it was expanded to two bits. To retain backwards compatibility in the model, `SEC_SID` remains as one-bit in the parameter `howto_identify`, but the second bit can be expressed with `SEC_SID_bit_1`, and its negative logic version `nSEC_SID_bit_1`.

Table 3-1178: Security State Determination (SSD) encoding

Security state	SSD	SEC_SID_bit_1	SEC_SID
Secure	0	0	1
Non-secure	1	0	0
Root (reserved)	2	1	1

Security state	SSD	SEC_SID_bit_1	SEC_SID
Realm	3	1	0

Ports for SMMUv3AEM

Table 3-1179: Ports

Name	Protocol	Type	Description
axi_stream_msi_addr_to_match_s	Value_64	Slave	Any SMMU-originated MSI which exactly matches the address in this port will be sent through the AXI stream port axi_stream_msi_m which is usually connected to the GIC through axi_stream_msi_s. As MSIs are 32 bit aligned then if bits[1:0] != 0 or the address is above OAS then this is effectively disabled. The parameter smmu_msi_device_id is the DeviceID to send on the interface. See also the parameters: axi_stream_msi_TID and axi_stream_msi_TDEST. The default value of this port is set by the parameter axi_stream_msi_addr_to_match. This port is sampled at negedge of reset_in and must be set before the negedge of the reset signal.
axi_stream_msi_m	PVBus	Master	Manager port used for sending SMMU originated MSIs directly to the GIC when axi_stream_msi_enabled == true
clk_in	ClockSignal	Slave	Clock signal
conf_reset_of_SMMU_GBPA_ABORT	Signal	Slave	System reset value of SMMU_GBPA.ABORT. This port is sampled at negedge of reset_in and must be set before the negedge of the reset signal.
conf_reset_of_SMMU_S_GBPA_ABORT	Signal	Slave	System reset value of SMMU_S_GBPA.ABORT. This port is sampled at negedge of reset_in and must be set before the negedge of the reset signal.
conf_system_supports_bgptm	Signal	Slave	System supports broadcast TLBI PAALL and TLBI RPA for supporting RME. This port is sampled at negedge of reset_in and must be set before the negedge of the reset signal.
conf_system_supports_btm	Signal	Slave	System supports BTM and will be reflected in the IDR registers. This signal can override the value set by the parameters configuring the IDR registers. If BTM (Broadcast Table Maintenance) is not supported then DVM messages will be ignored. This port is sampled at negedge of reset_in and must be set before the negedge of the reset signal.
conf_system_supports_cohacc	Signal	Slave	System supports COHACC and will be reflected in the IDR registers. This signal can override the value set by the parameters configuring the IDR registers. If COHACC is set then page walks and SMMU-generated accesses will have the required shareability set, otherwise they will be marked as non-shareable. This port is sampled at negedge of reset_in and must be set before the negedge of the reset signal.
conf_system_supports_httu	Signal	Slave	System supports HTTU and will be reflected in the IDR registers. See parameter support_for_httu_when_starts_disallowed for the use of this signal. This port is sampled at negedge of reset_in and must be set before the negedge of the reset signal.

Name	Protocol	Type	Description
conf_system_supports_sev	Signal	Slave	System supports SEV and will be reflected in the IDR registers. This signal can override the value set by the parameters configuring the IDR registers. This port is sampled at negedge of reset_in and must be set before the negedge of the reset signal.
identify	SMMUv3AEMIdentifyProtocol	Master	Map the transaction to the tuple (StreamID, SubstreamID, SubstreamIDValid, SSD)
irq_out_command_queue_sync_ns	Signal	Master	Pulsed interrupt output signal for non-secure CMD_SYNC having a completion signal of SIG_IRQ.
irq_out_command_queue_sync_rl	Signal	Master	Pulsed interrupt output signal for realm CMD_SYNC having a completion signal of SIG_IRQ.
irq_out_command_queue_sync_s	Signal	Master	Pulsed interrupt output signal for secure CMD_SYNC having a completion signal of SIG_IRQ.
irq_out_event_queue_ns	Signal	Master	Pulsed interrupt output signal for the non-secure event queue becoming non-empty.
irq_out_event_queue_rl	Signal	Master	Pulsed interrupt output signal for the realm event queue becoming non-empty.
irq_out_event_queue_s	Signal	Master	Pulsed interrupt output signal for the secure event queue becoming non-empty.
irq_out_gerror_ns	Signal	Master	Pulsed interrupt output signal for non-secure SMMU_GERROR(N) signaling an error.
irq_out_gerror_rl	Signal	Master	Pulsed interrupt output signal for realm SMMU_GERROR(N) signaling an error.
irq_out_gerror_s	Signal	Master	Pulsed interrupt output signal for secure SMMU_GERROR(N) signaling an error.
irq_out_gpf_far	Signal	Master	For RME-enabled SMMUs. A new error reported in SMMU_ROOT_GPF_FAR will pulse this interrupt.
irq_out_gpt_cfg_far	Signal	Master	For RME-enabled SMMUs. A new error reported in SMMU_ROOT_GPT_CFG_FAR will pulse this interrupt.
irq_out_ns	Signal	Master	Pulsed interrupt output signal combined from all non-secure (non-RAS) interrupts. This delivers a pulse if any of the interrupt pins of the specified world also deliver a pulse. SW will have to poll all the different reasons to see why it was delivered.
irq_out_pmcg_ns_as_value	Value	Master	Non-secure PMCG interrupt value port. Value port representing a set of Performance Monitor Counter Group interrupts. There is an unknown number of PMCGs and so an unknown number of PMCG interrupts. There may not necessarily even be an interrupt per group. We export the interrupt to be generated as a unsigned. The 'pmcg_index' is exported on the top 16 bits and the 'pmcg_counter' (that overflowed) on the bottom 16 bits. Any configured MSI for this group will be the next transaction out of the pvbus_m_tw_msi_qs port. The architecture supports an MSI from a PMCG could come out of a different port (say the TBU manager port that a PMCG might be associated with). However the AEM only supports it coming out of the TCU port (pvbus_m_tw_msi_qs).

Name	Protocol	Type	Description
irq_out_pmcg_s_as_value	Value	Master	Secure PMCG interrupt value port. Value port representing a set of Performance Monitor Counter Group interrupts. There is an unknown number of PMCGs and so an unknown number of PMCG interrupts. There may not necessarily even be an interrupt per group. We export the interrupt to be generated as a unsigned. The 'pmcg_index' is exported on the top 16 bits and the 'pmcg_counter' (that overflowed) on the bottom 16 bits. Any configured MSI for this group will be the next transaction out of the pvbms_mtw_msi_qs port. The architecture supports an MSI from a PMCG could come out of a different port (say the TBU manager port that a PMCG might be associated with). However the AEM only supports it coming out of the TCU port (pvbms_mtw_msi_qs).
irq_out_pri_queue	Signal	Master	Pulsed interrupt output signal for the non-secure PRI queue.
irq_out_pri_queue_rl	Signal	Master	Pulsed interrupt output signal for the realm PRI queue.
irq_out_ras_cri_as_value	Value	Master	RAS Critical error interrupt value port. The number of RAS error record groups is dependent on the configuration. As each error record group could have its own interrupts then we write to this value port the error record group number (which is dependent on the configuration) to simulate an interrupt pulse. We only support modelling edge-triggered interrupts. The value passed is formatted as follows: bits[23:16] the ras error group index (if multiple error groups) (0 atm) bits[15:8] node id bits[7:0] the ras record index in the node.
irq_out_ras_eri_as_value	Value	Master	RAS Error Handling Interrupt value port. The number of RAS error record groups is dependent on the configuration. As each error record group could have its own interrupts then we write to this value port the error record group number (which is dependent on the configuration) to simulate an interrupt pulse. We only support modelling edge-triggered interrupts. The value passed is formatted as follows: bits[23:16] the ras error group index (if multiple error groups) (0 atm) bits[15:8] node id bits[7:0] the ras record index in the node.
irq_out_ras_fhi_as_value	Value	Master	RAS Fault Handling Interrupt value port. The number of RAS error record groups is dependent on the configuration. As each error record group could have its own interrupts then we write to this value port the error record group number (which is dependent on the configuration) to simulate an interrupt pulse. We only support modelling edge-triggered interrupts. The value passed is formatted as follows: bits[23:16] the ras error group index (if multiple error groups) (0 atm) bits[15:8] node id bits[7:0] the ras record index in the node.
irq_out_rl	Signal	Master	Pulsed interrupt output signal combined from all realm (non-RAS) interrupts. This delivers a pulse if any of the interrupt pins of the specified world also deliver a pulse. SW will have to poll all the different reasons to see why it was delivered.
irq_out_s	Signal	Master	Pulsed interrupt output signal combined from all secure (non-RAS) interrupts. This delivers a pulse if any of the interrupt pins of the specified world also deliver a pulse. SW will have to poll all the different reasons to see why it was delivered.

Name	Protocol	Type	Description
logptsiz_s	Value	Slave	RME: This is a four bit signal that encodes the region size that a single LOGPT entry covers. The default value of this port is derived from the parameter <code>rme_logpt_entry_covers_log2size_in_bytes</code> which is in a different format to the port. If a valid value is driven then it will be put in the field <code>SMMU_ROOT_GPT_BASE_CFG.LOGPTSZ</code> . The port uses the same encoding as the field. If an invalid value is driven to this pin and <code>legacy_tz_en</code> is low then the model will obey the setting of the parameter <code>out_of_range_logptsiz_s</code> . This port is sampled at negedge of <code>reset_in</code> and must be set before the negedge of the reset signal.
legacy_tz_en	Signal	Slave	For an RME-enabled SMMU then tie this high to get non-RME behaviour. See also the parameter <code>SMMU_ROOT_IDRO</code> . This port is sampled at negedge of <code>reset_in</code> and must be set before the negedge of the reset signal.
pvbus_control_s	PVBus	Slave	Register subordinate port
pvbus_id_routed_m	PVBus	Master	This is a special "id-routed" port for transmitting ATC invalidates and PRI Responses upstream into the PCIe EndPoints, it is not a normal bus. The FastSim ATC invalidate protocol specifies how to route and deal with this this port. See the parameter <code>output_id_routed_transform</code> . It is assumed that the StreamID can uniquely route the transaction if there are multiple PCIe Root Complexes.
pvbus_m[64]	PVBus	Master	The TBU manager ports that carry transactions that have been translated from the correspondingly numbered <code>pvbus_s[]</code> port.
pvbus_m_tw_msi_qs	PVBus	Master	Manager port used for Table Walks, MSIs and Queue access when <code>separate_tw_msi_qs_port=true</code>
pvbus_s[64]	PVBus	Slave	The TBU subordinate ports that receive transactions to be translated. They will exit the SMMU through the same numbered <code>pvbus_m[]</code> port.
reset_in	Signal	Slave	Reset signal
sev_out	Signal	Master	Event signal

Parameters for SMMUv3AEM

PRESET_REL_base_address

If using preset addresses (`SMMU_IDR1.QUEUES_PRESET/TABLES_PRESET`) then the queue and table base registers become fixed. If `SMMU_IDR1.REL` then the addresses are relative to the base of the register file and this parameter tells the model what address to add to the queue/table addresses to calculate the actual address.

This is for 'embedded implementations' where the memory for these structures is held within the SMMU itself or in a 'close' RAM. The model does not contain any RAM and the integrator must supply a RAM at the appropriate address.

If the preset tables/queues overlap, the RAM has to implement separate secure and non-secure address spaces.

See also: `* TABLES_PRESET_smmu_{s,r}_strtab_base *`
`TABLES_PRESET_smmu_{s,r}_strtab_base_cfg * QUEUES_PRESET_smmu_{s,r}_cmdq_base`

* QUEUES_PRESET_smmu_{s,r}eventq_base * QUEUES_PRESET_smmu_{r}priq_base (no secure PRIQ)

Type

int

Default value

0x0

QUEUES_PRESET_smmu_cmdq_base

If SMMU_IDR1.QUEUES_PRESET == 1 then this is the value that appears in SMMU_CMDQ_BASE and SMMU_CMDQ_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

QUEUES_PRESET_smmu_eventq_base

If SMMU_IDR1.QUEUES_PRESET == 1 then this is the value that appears in SMMU_EVENTQ_BASE and SMMU_EVENTQ_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

QUEUES_PRESET_smmu_priq_base

If SMMU_IDR1.QUEUES_PRESET == 1 and SMMU_IDR0.PRI == 1 then this is the value that appears in SMMU_PRIQ_BASE and SMMU_PRIQ_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

QUEUES_PRESET_smmu_r_cmdq_base

If SMMU_IDR1.QUEUES_PRESET == 1 then this is the value that appears in SMMU_R_CMDQ_BASE and SMMU_R_CMDQ_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

QUEUES_PRESET_smmu_r_eventq_base

If SMMU_IDR1.QUEUES_PRESET == 1 and SMMU_ROOT_IDR0.REALM_IMPL == 1 then this is the value that appears in SMMU_R_EVENTQ_BASE and SMMU_R_EVENTQ_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

QUEUES_PRESET_smmu_r_priq_base

If SMMU_IDR1.QUEUES_PRESET == 1 and SMMU_IDR0.PRI == 1 and SMMU_ROOT_IDR0.REALM_IMPL == 1 then this is the value that appears in SMMU_PRIQ_BASE and SMMU_PRIQ_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

QUEUES_PRESET_smmu_s_cmdq_base

If SMMU_IDR1.QUEUES_PRESET == 1 and SMMU_S_IDR1.SECURE_IMPL == 1 then this is the value that appears in SMMU_S_CMDQ_BASE and SMMU_S_CMDQ_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

QUEUES_PRESET_smmu_s_eventq_base

If SMMU_IDR1.QUEUES_PRESET == 1 and SMMU_S_IDR1.SECURE_IMPL == 1 then this is the value that appears in SMMU_S_EVENTQ_BASE and SMMU_S_EVENTQ_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

SMMU_AIDR

SMMU_AIDR contains the Major and Minor architectural revisions numbers

Type

int

Default value

0x1

SMMU_IDR0

SMMU_IDR0. The following fields are further combined with the port
conf_system_supports_{sev,httu,btm,cohacc}:- * sev * htту * btm * cohacc

NOTE that SMMU_IDR0.RME_IMPL is the value that the SMMU should have if the SMMU is currently RME-aware. It will be forced to zero if the SMMU has been forced to be unaware of RME by legacy_tz_en.

Type

int

Default value

0x81ff6bf

SMMU_IDR1

SMMU_IDR1.

Type

int

Default value

0xe739d20

SMMU_IDR2

SMMU_IDR2 holds the BA_VATOS field.

Type

int

Default value

0x1

SMMU_IDR3

SMMU_IDR3 is reserved.

Type

int

Default value

0x71c

SMMU_IDR4

SMMU_IDR4 is Imp def.

Type

int

Default value

0x0

SMMU_IDR5

SMMU_IDR5 contains, amongst others the output address encoded size (OAS).

Type

int

Default value

0xffff0476

SMMU_IDR6

SMMU_IDR6 is RES0 if Enhanced Command Queues do not exist (SMMU_IDR1.ECMDQ == 0).

Otherwise, SMMU_IDR6 contains information about the configuration of the ECMDQs.

Type

int

Default value

0x0

SMMU_IIDR

SMMU_IIDR contains fields for the implementer, product revision, etc.

Type

int

Default value

0x0

SMMU_MPAMIDR

SMMUv3.2: If SMMU_IDR3.MPAM == 1 then SMMU_MPAMIDR holds further ID information for Memory Partitioning And Monitoring (MPAM) extension.

This is optional in SMMUv3.2 and is backported to SMMUv3.1.

Type

int

Default value

0x0

SMMU_ROOT_IDR0

If SMMU_ROOT_IDR0 is 0 then the SMMU is RME-unaware.

Otherwise...

legacy_tz_en is a pin that when high disables RME and the SMMU_ROOT_IDR0 register reads as zero.

The effective value of legacy_tz_en is derived from * the last signalled value sampled at negedge of reset * or if never signalled, the inverse of ROOT_IMPL (bit[0]) of this parameter.

Thus, ROOT_IMPL should be zero if we want legacy_tz_en to start as high regardless of the actual configuration we want in the SMMU_ROOT_IDR0 register when the SMMU is RME-aware.

In other words, if the SMMU is to be RME-aware, then all parameters should be configured as though the SMMU is currently RME-aware with the exception that SMMU_ROOT_IDR0.ROOT_IMPL is the inverse of the default value of legacy_tz_en.

SMMU_ROOT_IDR0.BGPTM is the default value of the pin conf_system_supports_bgptm.

Type

int

Default value

0x0

SMMU_ROOT_IIDR

The value of the SMMU_ROOT_IIDR register. If is zero then will be the same as SMMU_IIDR.

Type

int

Default value

0x0

SMMU_R_AIDR

The value of SMMU_R_AIDR.

Type

int

Default value

0x0

SMMU_R_IDR0

The value of SMMU_R_IDR0.

Type

int

Default value

0x0

SMMU_R_IDR3

The value of SMMU_R_IDR3.

Type

int

Default value

0x0

SMMU_R_IDR6

The value of SMMU_R_IDR6 that configures the ECMDQs.

Type

int

Default value

0x0

SMMU_R_MECIDR

The value of SMMU_R_MECIDR.

Type

int

Default value

0x0

SMMU_R_MPAMIDR

NOTE this parameter is 64 bits but the ID register is 32 bits.

This parameter is the value of SMMU_R_MPAMIDR, or if ~0ull then it will have the following default values:

* PARTID_MAX/PMG_MAX from the SMMU_MPAMIDR * HAS_MPAM_NS from the SMMU_S_MPAMIDR if it exists, otherwise 0.

Type

int

Default value

0xffffffffffffffff

SMMU_S_IDR0

Secure IDR0 register.

Type

int

Default value

0x2000

SMMU_S_IDR1

SMMU_S_IDR1 Indicates if there is a secure side by bit 31.

Type

int

Default value

0xa0000020

SMMU_S_IDR2

SMMU_S_IDR2 Reserved

Type

int

Default value

0x0

SMMU_S_IDR3

SMMU_S_IDR3 Reserved

Type

int

Default value

0x0

SMMU_S_IDR4

SMMU_S_IDR4 IMP DEF

Type

int

Default value

0x0

SMMU_S_IDR6

SMMU_S_IDR6 is RES0 if Secure Enhanced Command Queues do not exist (SMMU_S_IDR0.ECMDQ == 0).

Otherwise, SMMU_S_IDR6 contains information about the configuration of the ECMDQs.

Type

int

Default value

0x0

SMMU_S_MPAMIDR

SMMUv3.2: If SMMU_IDR3.MPAM == 1 then SMMU_S_MPAMIDR holds further ID information for Memory Partitioning And Monitoring (MPAM) extension.

This is optional in SMMUv3.2 and is backported to SMMUv3.1.

Type

int

Default value

0x0

TABLES_PRESET_smmu_r_strtab_base

If SMMU_IDR1.TABLES_PRESET == 1 and SMMU_ROOT_IDR0.REALM_IMPL == 1 then this is the value that appears in SMMU_R_STRTAB_BASE and SMMU_R_STRTAB_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

TABLES_PRESET_smmu_r_strtab_base_cfg

If SMMU_IDR1.TABLES_PRESET == 1 and SMMU_ROOT_IDR0.REALM_IMPL == 1 then this is the value that appears in SMMU_R_STRTAB_BASE_CFG and SMMU_R_STRTAB_BASE_CFG becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

TABLES_PRESET_smmu_s_strtab_base

If SMMU_IDR1.TABLES_PRESET == 1 and SMMU_S_IDR1.SECURE_IMPL == 1 then this is the value that appears in SMMU_S_STRTAB_BASE and SMMU_S_STRTAB_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

TABLES_PRESET_smmu_s_strtab_base_cfg

If SMMU_IDR1.TABLES_PRESET == 1 and SMMU_S_IDR1.SECURE_IMPL == 1 then this is the value that appears in SMMU_S_STRTAB_BASE_CFG and SMMU_S_STRTAB_BASE_CFG becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

TABLES_PRESET_smmu_strtab_base

If SMMU_IDR1.TABLES_PRESET == 1 then this is the value that appears in SMMU_STRTAB_BASE and SMMU_STRTAB_BASE becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

TABLES_PRESET_smmu_strtab_base_cfg

If SMMU_IDR1.TABLES_PRESET == 1 then this is the value that appears in SMMU_STRTAB_BASE_CFG and SMMU_STRTAB_BASE_CFG becomes read-only.

See also parameter PRESET_REL_base_address.

Type

int

Default value

0x0

all_error_messages_through_trace

Some conditions in the SMMU are so strange that the software programming the SMMU has done something wrong. At this point messages are output to either ArchMsg.Error.* or ArchMsg.Warning.* or to the error stream of the simulator. Outputting to the error stream of the simulator may cause it to return with a non-zero exit status.

If you set this option to true then instead of using the error stream of the simulator it will always use a trace stream allowing the simulation to exit with a zero exit status.

Type

bool

Default value

0x0

allow_non_secure_access_to_SMMU_S_INIT

If the system has no software operating as a secure agent then set this parameter. This allows non-secure accesses to the SMMU_S_INIT register and allows the non-secure software to reset the TLB, clearing out any 'secure' TLB entries.

If the SMMU does not implement the security extensions (SMMU_S_IDR1.SECURE_IMPL == 0) then this parameter is ignored.

Type

bool

Default value

0x0

apply_ste_instcfg_privcfg_on_all_ats_translated_accesses

If SMMU_IDR1.ATTR_PERMS_OVR == 0 then this parameter is ignored.

Otherwise, if this parameter is:

* false: STE.INSTCFG/PRIVCFG will only be applied to ATS-TranslatedTransactions if STE.EATS==split-stage.

* true: STE.INSTCFG/PRIVCFG will be applied to all ATS-TranslatedTransactions regardless of the value of STE.EATS.

Type

bool

Default value

0x0

ats_split_stage_dbm_update_do_with_ATSRequest

When doing split-stage ATS, then the DBM update for the final stage 2 descriptor can be done either whilst processing the ATS request or delayed until it actually sees the PCIe Translated Transaction using the stage 2 descriptor.

0 -- do when see actual transaction 1 -- do when processing the ATS request 2 -- do it randomly with 50% chance.

Type

int

Default value

0x0

axi_stream_msi_TDEST

ID of the AXI stream subordinate port on the GIC that receives SMMU originated MSIs sent directly. The name of the GIC port is axi_stream_msi_s.

NOTE that if axi_stream_msi_addr_to_match[1:0] != 0, or the address is above OAS then this is effectively disabled.

See also: * axi_stream_msi_addr_to_match * axi_stream_msi_TID

Type

int

Default value

0x0

axi_stream_msi_TID

ID of the AXI stream manager port that sends SMMU TCU originated MSIs directly to the GIC. The name of the SMMU port is axi_stream_msi_m.

NOTE that if axi_stream_msi_addr_to_match[1:0] != 0, or the address is above OAS then this is effectively disabled.

See also: * axi_stream_msi_addr_to_match * axi_stream_msi_TDEST

Type

int

Default value

0x0

axi_stream_msi_addr_to_match

If the last two bits are 0, any SMMU-originated MSI which exactly matches the address will be sent through the AXI stream port axi_stream_msi_m which is usually connected to the GIC.

This parameter drives the value of the axi_stream_msi_addr_to_match_s port at simulation reset. For every reset after that, the value of the port will be sampled and used if changed.

NOTE that the entire address must match, including bits [1:0]. As MSIs are 32 bit aligned then if axi_stream_msi_addr_to_match[1:0] != 0, or the address is above OAS then this is effectively disabled.

See also: * axi_stream_msi_TID * axi_stream_msi_TDEST

Type

int

Default value

0xffffffffffffffff

behaviour_of_sampled_at_reset_signals

Some configuration signals into the SMMU are sampled on negedge of reset.

However, it can sometimes be hard to arrange to drive a configuration pin before the negedge of reset.

The configuration pins are sampled: 0 -- at negedge reset. 1 -- at negedge reset, but if a later change occurs at the same simulated time, and no transactions have occurred, then they will be resampled and the SMMU reset again.

Type

int

Default value

0x0

cmdq_max_number_of_commands_to_buffer

The command queues can buffer fetched commands before issuing them. This parameter is roughly the maximum number of commands to do this for. The programmer visible effects are that just because the CONS pointer shows a command has been `_consumed_` does not necessarily mean that it has been issued (and completed). Higher values will accentuate this effect.

Type

int

Default value

0xa

dpt_configure_ATS_formed_entries

For ATS-formed DPT TLB entries then a comma-separated list of options. Some options are mutually exclusive.

If the string is empty or "none", then no ATS-formed entries are inserted into the DPT.

Otherwise...

One of:

* "all_enabled_stages" / "" -- use all enabled VMSA stages (default) * "last_enabled_stage" -- use only the last enabled VMSA stage.

The region stored in the DPT entry might be just the VMSA region or the VMSA region truncated by the GPC checked region. One of:

* "vmsa_and_gpc" / "" -- store as the smaller of the VMSA and GPC region (default) * "vmsa_only" -- store as the VMSA region

The implementation may distinguish ATS-formed entries and DPT-formed entries:

* "on_fault_always_walk" / "" -- ATS-formed and DPT-formed entries are not distinguished (default) * "on_fault_walk_if_ATS_formed" -- ATS-formed entries are not definitive and will cause a walk. DPT-formed entries are definitive and will not cause a walk.

Example:

"all_enabled_stages, vmsa_and_gpc"

Type

string

Default value

all_enabled_stages, vmsa_and_gpc

dpt_configure_invalidation

Configure when entries are invalidated when performing a DPT check.

A comma-separated list of options.

Choose one of: * lookup_fault_invalidates_any_existing_entries / "" (default) * lookup_fault_leaves_any_existing_entries

Choose one of: * noaccess_fault_invalidates_any_existing_entries / "" (default) * noaccess_fault_leaves_any_existing_entries

Type

string

Default value

enable_device_id_checks

If this parameter is true then the DeviceIDs seen by the GIC are: * for client devices: DeviceID = StreamID + translated_device_id_base * for SMMU-generated MSIs: smmu_msi_device_id

This parameter enables two checks:

* If the DeviceID is used in the output_attribute_transform then if it overflows 32 bits then the model will warn. If the DeviceID is not used then it is assumed that the external agent that forms the DeviceID will warn if it overflows.

* If the SMMU supports MSIs, then the model will check that the GIC will be able to distinguish an MSI generated by the SMMU from one generated by a client device.

As the exact mechanism to determine the DeviceID is in the system and not necessarily under control of the SMMU then you can disable these warnings using this parameter.

See also the parameters: output_attribute_transform and msi_attribute_transform.

Type

bool

Default value

0x0

hide_warning_EOPD_differs_from_what_would_be_cached

When this parameter is set to true, warnings that the effective EOPD value differs from what would be cached in the TLB are disabled. False (warnings are showed) by default.

Type

bool

Default value

0x0

hide_warning_NoStreamID_transaction_for_unsupported_PAS_or_MPAM_SP

When RME is not supported then a NoStreamID transaction with PAS[1] == 1 or MPAM_SP[1] == 1 is treated as though PAS[1] == 0 and MPAM_SP[1] == 0. This is usually a system construction error and is not expected to occur.

The SMMU will warn when this occurs, but the warning can be hidden by setting this parameter.

Type

bool

Default value

0x0

howto_identify

If 'use-identify' then will use the 'identify' port to determine the SSD, StreamID, SubStreamID. Otherwise, this string extracts them from the transaction's attributes.

Examples:-

```
SEC_SID=ExtendedID[63], SSV=ExtendedID[62], SubstreamID=ExtendedID[51:32],
StreamID=ExtendedID[31:0] nSEC_SID=ExtendedID[63], StreamID=ExtendedID[55:24],
nSSV=ExtendedID[20], SubstreamID=ExtendedID[19:0] StreamID[31:24]=0,
StreamID[23:0]=ExtendedID[23:0], SSV=1[0], ...
```

The StreamID (32 b) is valid if SIDV is 1 or both SIDV and nSIDV are unused.

SEC_SID is 1b in the model. For RME systems, in HW then SEC_SID is 2b, in the model SEC_SID_bit_1 represents bit[1].

The SubstreamID (20 b) is valid if SSV is true.

SIDV == 0 indicates a NoStreamID transaction and the SSD is the PAS of the transaction, SEC_SID is not used.

nSEC_SID, nSSV, nSIDV are available with negative logic. Different attributes are independent and can use negative or positive logic.

Type

string

Default valueSEC_SID=ExtendedID[63], SSV=ExtendedID[62], SubstreamID=ExtendedID[51:32],
StreamID=ExtendedID[31:0]**howto_identify_NoStreamID_extra_info**

The behavior of this parameter depends on 'howto_identify' * if it equals 'use-identify' then this must be "", otherwise there is an error. * if it identifies a NoStreamID transaction (SIDV=0) then this parameter includes one or more of * MPAM_SP * MPAM_PARTID * MPAM_PMG * MECID * HWATTR_KIND_0 * in any other case, this parameter is ignored. Fields set in this parameter MUST NOT overlap the SIDV/nSIDV fields in 'howto_identify'

Example:-

MPAM_PMG[7:0]=ExtendedID[62:55], MPAM_PARTID[15:0]=ExtendedID[54:39],
MPAM_SP[1:0]=ExtendedID[38:37], MECID[15:0]=UserFlags[31:16]
HWATTR_KIND_0[3:0]=ExtendedID[42:39]

Type

string

Default value**httu_early_st2_permission_fault_if_af_update_at_stagel**

If a stage 1 descriptor needs an HTTU update, but the descriptor is unwriteable at stage 2 and also a stage 1 permission fault occurs, then the architecture permits either the stage 1 or stage 2 permission fault to be recorded.

0 -- stage 1 permission check check stage 1 descriptor writeable at stage 2 if AF- or DBM-update required

1 -- check stage 1 descriptor writeable at stage 2 if AF-update required stage 1 permission check check stage 1 descriptor writeable at stage 2 if DBM-update required

2 -- do behaviour 1 or 2 randomly with a 50% chance.

Type

int

Default value

0x0

httu_memory_types_supported

This is a comma-separated list of memory types that are IMPLEMENTATION DEFINED as supporting HTTU. However, the system must have Far Atomic support for the specified memory address and memory type.

Device types: * nGnRnE, nGnRE, nGRE, GRE

Normal memory types are composed of an 'inner' and an 'outer' cacheability. The model only supports types where the inner and outer are identical.

* Normal non-cacheable types * nc_nb, nc * Cacheable types are of the form (na?|(ra?)(wa?)(WT|WB)(tr)? * na/ra/wa -- no/read/write allocate * WT/WB -- write through/write back * tr -- transient * exceptions: * 'na' and 'tr' are incompatible * without 'na' then you must specify at least one of ra/wa. Example: "WT" is illegal, "raWT" is legal.

rawaWB is always supported and it is optional.

Examples: * "rawaWB, raWB, waWB, naWB" -- only the WB type is supported * "nc" -- rawaWB and the normal non-cacheable type are supported

Type

string

Default value

rawaWB, raWB, waWB, naWB

imp_def_L1CD_L2Ptr_out_of_range

If an L1CD.L2Ptr is out of range of IAS/OAS as appropriate then what happens is controlled by this parameter: 0 -- if is an IPA, then Stage 2 Translation Fault, if is a PA then truncate to OAS 1 -- generate C_BAD_SUBSTREAMID if an IPA and > IAS, or if a PA and > OAS. 2 -- generate C_BAD_SUBSTREAMID if an IPA and > IAS, or F_CD_FETCH if a PA and > OAS. 3 -- truncate the IPA or PA to IAS/OAS as appropriate

NOTE that if the model is configured as SMMUv3.1 then this parameter is IGNORED, and behaves as though this parameter was set to 1. The SMMUv3.1 architecture actually allows more behaviours but the model will only implement this one.

NOTE that the SMMUv3.0 allows more behaviours than can be expressed by this parameter.

Type

int

Default value

0x0

imp_def_PID0

If imp_def_has_PID_CID is true then this is the PID0 value.

Type

int

Default value

0x83

imp_def_PID1

If imp_def_has_PID_CID is true then this is the PID1 value.

Type

int

Default value

0xb4

imp_def_PID2

If imp_def_has_PID_CID is true then this is the PID2 value.

Type

int

Default value

0xb

imp_def_PID3

If imp_def_has_PID_CID is true then this is the PID3 value.

Type

int

Default value

0x0

imp_def_PID4

If imp_def_has_PID_CID is true then this is the PID4 value.

Type

int

Default value

0x4

imp_def_S1ContextPtr_out_of_range

If an STE is fetched that uses a stage 1 then if: - Stage 1 only and S1ContextPtr > OAS, or
 - Stage 1+2 and S1ContextPtr > IAS then what happens is IMP DEF and this parameter controls the behaviour:- 0 -- stage 1 only -- C_BAD_STE -- stage 1+2 -- C_BAD_STE 1 -- stage 1 only -- C_BAD_STE -- stage 1+2 -- truncate to IAS 2 -- stage 1 only -- truncate to OAS -- stage 1+2 -- C_BAD_STE 3 -- stage 1 only -- truncate to OAS -- stage 1+2 -- truncate to IAS 4 -- stage 1 only -- truncate to OAS -- stage 1+2 -- Stage 2 translation fault 5 -- stage 1 only -- C_BAD_STE -- stage 1+2 -- Stage 2 translation fault The architecture also allows for F_CD_FETCH, but the model does not support this.

NOTE that in SMMUv3.1 then the only allowed values of this parameter are 0 or 5.

Type

int

Default value

0x0

imp_def_alloccfg

ALLOCCFG overrides the read/write/transient hints on cacheable types. However these are hints and an implementation may choose to treat them differently. 0 -- apply the alloc hints as architecturally specified 1 -- ignore all ALLOCCFG fields (treated as zero) 2 -- apply ALLOCCFG only when MTCFG == 1

Type

int

Default value

0x0

imp_def_apply_dre_dcp_to_full_ats

STE.DRE and STE.DCP control the downgrade of certain transactions to NOP.

If this parameter is true, then for ATS-TranslatedTransactions using STE.EATS == eats_all_stages then the STE.DRE/STE.DCP controls are applied.

For ATS-TranslatedTransactions using STE.EATS == eats_use_dpt then the controls are always applied.

For SMMUv3.4 and later, or if SMMU_IDR3.DPT == 1 then this parameter is ignored and treated as 'true'.

Type

bool

Default value

0x0

imp_def_ats_attribute_stashing

The SMMU architecture allows an ATS request to return the attributes with which to make the Translated Access. PCIe does not define any transaction attributes in the ARM sense and so the mechanism for doing this is IMP DEF. Usually this would be done by packing them into the high order address bits of the return response.

In the model, then the representation of the ATS reply returns the attributes directly and it is up to the ATC whether it wants to use them or not.

The parameter configures what to place in those architectural attributes in the ATS Reply.

0 -- the architectural attributes 1 -- Inner Write Back, Outer Write Back, Inner Shared, read and write allocate, User-Data 2 -- Inner Write Back, Outer Write Back, Outer Shared, read and write allocate, User-Data

The SMMU cannot force an ATC to use these attributes, it is simply the attributes that are returned in the non-PCIe part of the ATS reply.

Type

int

Default value

0x0

imp_def_ats_response_stu

A successful ATS Response with RW != 0 can return any-sized region from the STU to the actual region size. The chosen size is:

* 0 -- use the full size of the region * 1 -- use the STU * 2 -- use half the size of the region

Type

int

Default value

0x0

imp_def_cohacc_effect

SMMU_IDR0.COHAAC is a system property. However, the exact nature of the transactions that the SMMU emits is an IMP DEF property when COHAAC == 0: 0 -- COHAAC == 0 forces the output attributes of SMMU-generated accesses to non-shared. 1 -- The only effect of COHAAC is what is reported in SMMU_IDR0.COHAAC and has no effect on the output attributes of SMMU-generated accesses.

Type

int

Default value

0x0

imp_def_contiguous_bit_handling

If the Contiguous bit is set in a translation table descriptor then modify how it is cached:

* 0 -- use the full size of the contiguous region * 1 -- ignore the contig bit for determining the region size * 2 -- use half the size of the contiguous region

See imp_def_gpt_contiguous_bit_handling for GPT Contig bit handling.

Type

int

Default value

0x0

imp_def_effective_ATTR_TYPES_OVR_is_false_per_port

SMMU_IDR1.ATTR_TYPES_OVR == 1 means that the STE and SMMU_(S_)GBPA MTCFG/SHCFG/ALLOCCFG have an effect.

However, an implementation is allowed to ignore this being one for specific ports and _not_ apply the overrides MTCFG/SHCFG/ALLOCCFG despite SMMU_IDR1.ATTR_TYPES_OVR == 1.

This parameter is a comma-separated lists of port ranges (indexed from 0) for those ports where SMMU_IDR1.ATTR_TYPES_OVR behaves as 0. For example:

0, 10-20, 40

Type

string

Default value**`imp_def_gpt_contiguous_bit_handling`**

If we find a GPT Contiguous descriptor then modify how it is cached:

* 0 -- use the full size of the contiguous region * 1 -- ignore the contig bit for determining the region size and use PGS * 2 -- use half the size of the contiguous region

See `imp_def_contiguous_bit_handling` for VMSA Contig bit handling.

Type

int

Default value

0x0

`imp_def_has_PID_CID`

If this is true then the SMMU model will have the standard PID/CID ID registers. Only the PID0..PID4 registers can be customized and the parameters `imp_def_PID0`..`imp_def_PID4` are used.

Type

bool

Default value

0x1

`imp_def_ns_bit_for_s_gatos_on_s1_bypass_non_sel2`

This parameter only has an effect if `SEL2 == 0`.

When `SEL2 == 0`, Secure virtualisation is not supported and only stage 1 is supported for Secure Streams.

In this case, the Secure ATOS interface does not provide a mechanism to specify the input NS bit to the stage 1 translation. The input bit is IMP DEF and only has an effect if the transaction has no SubstreamID and bypasses by S1DSS.

This parameter specifies the IMP DEF input bit: 0 -- secure 1 -- non-secure 2 -- random

See also `imp_def_ns_bit_for_s_vatos_on_s1_bypass` which configures something similar for the Secure VATOS (not GATOS) interface.

Type

int

Default value

0x0

imp_def_ns_bit_for_s_vatos_on_s1_bypass

When a Secure VATOS operations for a translation that bypasses stage 1 by S1DSS then the output NS bit is the same as the input NS bit of the translation.

The architecture does not provide an input NS bit in the SMMU_S_VATOS_ADDR register and it is treated as an IMP DEF value.

This parameter specifies that value: 0 -- secure 1 -- non-secure 2 -- random

See also `imp_def_ns_bit_for_s_gatos_on_s1_bypass_non_sel2` which configures something similar for the Secure GATOS (not VATOS) interface.

Type

int

Default value

0x0

imp_def_ras_access_control_policy

The access control of the RAS nodes:

* "" / "use-imp_def_ras_allow_non_secure_accesses_if_supports_secure" * The parameter `imp_def_ras_allow_non_secure_accesses_if_supports_secure` is used to determine access * "non-secure" * Allow access to all PAses. * "secure" * Only allow access to secure and root. If secure is not implemented then will allow access to non-secure. * "root-only" * The register file is only accessible to root-pas transactions. If root is not implemented then it will act as though this parameter was: "use-imp_def_ras_allow_non_secure_accesses_if_supports_secure"

Type

string

Default value

use-imp_def_ras_allow_non_secure_accesses_if_supports_secure

imp_def_ras_allow_non_secure_accesses_if_supports_secure

If two security worlds are supported, i.e.: `SMMU_S_IDR1.SECURE_IMPL == 1` then if this parameter is true, then non-secure accesses are allowed to access any RAS registers (see parameter 'ras'). Otherwise, non-secure accesses are RAZ/WI.

If only a single security state (non-secure) is supported, then this parameter is ignored and non-secure accesses are always allowed.

See also `imp_def_ras_access_control_policy` that can override this parameter.

Type

bool

Default value

0x0

imp_def_reset_unknown_fields_to_zero

Many fields and registers in the SMMUV3 architecture reset to an UNKNOWN value. However, many implementations will choose to reset to 0. By setting this parameter to true then those fields will be initialised to zero.

Type

bool

Default value

0x0

imp_def_rme_gpf_syndrome_for_PMCG_MSIs

An MSI access from a PMCG that experiences a GPF is permitted to be reported as either of:

* REASON = GERROR and FAULTCODE = OTHER_GPF * REASON = TRANSACTION

The values of this string are one of: * other_gpf * transaction

See also the parameter imp_def_rme_gpf_syndrome_for_RAS_MSIs

Type

string

Default value

other_gpf

imp_def_rme_gpf_syndrome_for_RAS_MSIs

An MSI access from a RAS record interrupt that experiences a GPF is permitted to be reported as either of:

* REASON = GERROR and FAULTCODE = OTHER_GPF * REASON = TRANSACTION

The values of this string are one of: * other_gpf * transaction

See also the parameter imp_def_rme_gpf_syndrome_for_PMCG_MSIs

Type

string

Default value

other_gpf

imp_def_rme_mpam_info_from_NoStreamID_on_gpt_walks_ignored

The MPAM related fields set in 'howto_identify_NoStreamID_extra_info' are ignored when this parameter is set. This parameter only makes sense when 'howto_identify' equals 'use-identify' so in any other case it must be false.

When this parameter is set: * MPAM_SP = PAS * MPAM_PARTID = 0 * MPAM_PMG = 0

Type

bool

Default value

0x0

imp_def_split_ATS_attributes_is_stagel

If using split stage ATS, then it is IMP DEF whether the stage 1 attributes are returned to the ATS request or stage 2.

This only has a meaning if the SMMU can stash attributes in the ATS response.

Type

bool

Default value

0x0

imp_def_truncate_out_of_range_streamids_on_invalidate_commands

If this parameter is true then the StreamID fields of the following commands will be truncated to (S_)SIDSIZE:

* CMD_ATC_INV * CMD_CFGI_STE * CMD_CFGI_STE_RANGE * CMD_CFGI_CD *
CMD_CFGI_CD_ALL

Otherwise, these commands will NOP.

Type

bool

Default value

0x0

imp_def_v3_atos_fault

For an IPA to PA ATOS translation that encounters a Stage 1 Address Size Fault then the PAR.REASON field reports: * in SMMUv3.1, 'Stage 1' (0) * in SMMUv3.0, 'Stage 1' (0) or 'Input' (3) depending on the implementation.

This parameter is ignored for SMMUv3.1.

For SMMUv3.0 then the values are: 0 -- report as 'Input' (3) 1 -- report as 'Stage 1' (0)

Type

int

Default value

0x0

ish_is_osh

When set, any transaction that would use the architectural inner shareable domain is converted to use the outer shareable domain.

NOTE that this parameter should match the equivalent `ish_is_osh` from the PE. If an incompatible value of the `ish_is_osh` parameter is configured for the PE and the SMMU, data coherency may be compromised.

Type

bool

Default value

0x0

mec_attribute_transform

If MEC is supported, this is applied to `_all_` downstream transactions to transport the MEC information. * ""/"none" -- no transform * How to alter the output attributes. Example: "UserFlags[31:16]=MECID[15:0]"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags.

RHS Symbols: * MECID * numeric literals.

Any bits with no transform are unchanged.

NOTE: * attribute transforms applied before this: * for client transactions 'output_attribute_transform'/'output_attribute_transform_for_NoStreamID'. * for table walks 'tw_qs_attribute_transform'. * for MSIs 'msi_attribute_transform'. * if MPAM is enabled 'mpam_attribute_transform'.

Type

string

Default value**mpam_attribute_transform**

If MPAM is supported, this is applied to `_all_` downstream transactions to transport the MPAM information. * ""/"none" -- no transform * How to alter the output attributes. Example: "ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID, ExtendedID[38]=MPAM_SP[0]"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags.

RHS Symbols: * MPAM_PARTID * MPAM_PMG * MPAM_NS * MPAM_SP * numeric literals

Any bits with no transform are unchanged.

NOTE: * attribute transforms applied before this: * for client transactions 'output_attribute_transform' / 'output_attribute_transform_for_NoStreamID'. * for table walks 'tw_qs_attribute_transform'. * for MSIs 'msi_attribute_transform'. * 'mec_attribute_transform' is applied after this. * for translated transactions from client devices then MPAM_NS = ! SEC_SID.

Type

string

Default value

ExtendedID[62:55]=MPAM_PMG, ExtendedID[54:39]=MPAM_PARTID,
ExtendedID[38]=MPAM_NS

msi_attribute_transform

Transform downstream attributes of MSI transactions. * ""/"none" -- no transform * How to alter output attributes of SMMU-generated MSIs. Example: "UserFlags[15:0]=smmu_msi_device_id[31:16], MasterID[15:0]=smmu_msi_device_id[15:0], ExtendedID=0"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags

RHS Symbols: * the parameter smmu_msi_device_id * 'interrupt_kind' * 0/1 -- EVENTQ s/ns * 2 -- PRIQ * 3/4 -- CMD_SYNC s/ns * 5/6 -- GERROR s/ns * 7/8 -- PMCG s/ns * 9/10/11 -- RAS FHI/ERI/CRI * 12/13 -- gpf_far/gpt_cfg_far * 14/15/16/17 -- Realm EVENTQ/PRIQ/CMDQ/GERROR * HWATTR_KIND_0: PBHA information * numeric literals.

ExtendedID/MasterID/UserFlags start with values {0, 0xFFFFFFFF, 0} respectively.

Any bits with no transform are unchanged.

This transform can be used to determine the DeviceID passed to the GIC to distinguish MSIs generated by the SMMU from those generated by client devices.

NOTE: * see also 'output_attribute_transform' and enable_device_id_checks.

Type

string

Default value

ExtendedID[31:0]=smmu_msi_device_id, MasterID=0xFFFFFFFF

msi_ra_wa_tr

A bitmap of the Read Allocation, Write Allocate and Transient hints for MSIs to cacheable memory: bit[0] Transient bit[1] Write Allocate bit[2] Read Allocate If not Write Allocate then it will be forced to Read Allocate as a limitation of AMBA.

Type

int

Default value

0x7

non_arch_incoming_stronger_than_iWB_oWB_forces_output_iNC_oNC_or_stronger

If not empty, then this enables a specific non-architectural behaviour on the comma-separated list of port indexes, or ranges. For example:

0, 10-20, 40

In the normal translation process, then the input attributes are usually replaced by the attributes from the page tables or SMMU_(S_)GBPA.

The behaviour is:

if incoming attributes are iWB-oWB use the architectural attributes else use the stronger of iNC-oNC-osh and the architectural attributes.

This is useful if the ports represent transactions from the PCIe subsystem and the PCIe devices output: * iWB-oWB if not No_Snoop -> output is architectural attributes * iNC-oNC-osh if No_Snoop -> output is iNC-oNC-osh or stronger

Type

string

Default value**normalize_input_normal_non_iWB_oWB_to_iNC_oNC_osh**

When set, use Inner Non Cacheable, Outer Non Cacheable, Outer Shareable for any upstream transaction that would use any of the following attributes: * Normal Non-cacheable Bufferable * Normal Non-cacheable Non-bufferable * Write-through

NOTE that this parameter should match the equivalent configuration from the PE. If an incompatible value of this parameter is configured for the PE and the SMMU, data coherency may be compromised.

Type

bool

Default value

0x0

number_of_ports

The number of port pairs that the SMMU has.

Type

int

Default value

0x1

nw_dcp_extra_drop_conditions

NW-DCP is a hint and can be dropped for any reason.

This is a comma-separated list of:

* "INST" -- NW-DCP is architecturally 'data' but by enabling this option then STE.INSTCFG is applied and it can fault due to SIF and, for rl-ssd, preventing instruction access to ns-PAS.

* "SIF-cached" -- NW-DCP needs any of rwx permissions to go downstream. If SIF has been cached into the TLB entry then it will have removed execute permission and so for an execute-only page the NW-DCP would be denied.

If you set this option then, for NW-DCP, the SMMU will behave as if SIF was cached.

The parameter `ordering_of_PAN_and_xn_by_ns_pas` can force TLB-caching of SIF and takes precedence over this option.

* "combined-st2" -- when considering the stage 2 permissions then they are first combined with any stage 1 permissions before applying the permission check.

* "combined-st1-st2" -- always fetches all stages and combine before applying the permissions check.

Type

string

Default value**`ordering_of_PAN_and_xn_by_ns_pas`**

Execution from ns-pas can be forbidden:

* for secure, this is controlled by SMMU_S_CRO.SIF. * for realm, this is mandatory. In the model, we call this RIF and is always cached in the TLB.

When PAN is interpreted as EPAN then whether SIF/RIF is applied before or after PAN can get different results for the direct permission model.

In the model, the following options are available:

0 -- PAN applied first, SIF not cached in the TLB, RIF cached in the TLB. 1 -- PAN applied first, SIF/RIF cached in the TLB. 2 -- SIF/RIF cached in the TLB, then PAN applied

NOTE that if you cache SIF in the TLB then all SIF faults are no longer traced separately as SIF faults but as permission faults -- which architecturally they are reported as. As RIF is always cached in the TLB, then they are not distinguished in the trace separately to permission faults.

Type

int

Default value

0x0

out_of_range_CMD_ATC_INV_Size

If CMD_ATC_INV.Size > 52 then the model is allowed to:- 0 -- raise CERROR_ILL 1 -- treat as NOP

The architecture also allows for an UNKNOWN invalidate size to be used as well but the model does not support this.

Type

int

Default value

0x0

out_of_range_logptsz_s

If the port logptsz_s is driven to an invalid value and that value is used then the following behaviors are possible:

* -2 -- report the incoming value in SMMU_ROOT_GPT_BASE_CFG.LOGPTSZ and all transactions will report a GPT Config Error if GPC checking is enabled. * -1 -- produce an error and make the model unusable (default) * invalid LOGPTSZ encoding -- report this value in SMMU_ROOT_GPT_BASE_CFG.LOGPTSZ and all transactions will report a GPT Config Error if GPC checking is enabled.

All other values are reserved and act as -1.

Type

int

Default value

0xffffffffffffffff

output_attribute_transform

Transform downstream attributes of StreamID transactions.

* ""/"none": no transform * How to alter output attributes. Example:

"ExtendedID[15:0]=DeviceID[15:0], UserFlags[31]=nSSV, UserFlags[19:0]=SubstreamID, MasterID[10]=MasterID[11], MasterID[11]=MasterID[10]"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags: incoming/outgoing attributes.

RHS Symbols: * DeviceID: StreamID + translated_device_id_base * StreamID/SubstreamID/SSV/SEC_SID * nSSV/nSEC_SID/: negative logic versions. * St1PBHA/St2PBHA: Page Based Hardware Attributes from leaf descriptors (zero if not used). * STE_IMPDEF1: STE[127:116] * HWATTR_KIND_0: PBHA information * numeric literals. * SIDV = 1, nSIDV = 0 (fixed values to indicate StreamID)

Any bits with no transform are unchanged.

NOTE: * 'mpam_attribute_transform' and 'mec_attribute_transform' are applied in order after this. * see also 'output_attribute_transform_for_NoStreamID' for NoStreamID transactions.

Type

string

Default value

ExtendedID[31:0]=DeviceID

output_attribute_transform_for_NoStreamID

Transform downstream attributes of NoStreamID transactions.

* ""/"none": no transform * How to alter output attributes. Example:

"ExtendedID[15:0]=0, UserFlags[31]=1, UserFlags[19:0]=0, MasterID[10]=MasterID[11], MasterID[11]=MasterID[10] MasterID[9:6]=HWATTR_KIND_0"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags: incoming/outgoing attributes.

RHS Symbols: * SIDV = 0, nSIDV = 1 (fixed values to indicate NoStreamID) * PAS * HWATTR_KIND_0 * numeric literals.

Any bits with no transform are unchanged.

NOTE: * 'mpam_attribute_transform' and 'mec_attribute_transform' are applied in order after this. * see also 'output_attribute_transform' for StreamID transactions.

Type

string

Default value

ExtendedID[31:0]=0, ExtendedID[32]=1

output_id_routed_transform

The SMMU generates the following ID-routed transaction on the pvbus_id_routed_m bus: * ATC Invalidate * PRI Response

This parameter expresses how the SMMU should express: * the StreamID * the Trusted (T) bit

The value is a comma-separated list of assignments:

Address[27:12]=StreamID[15:0], ExtendedID[60]=T, ExtendedID[15:0]=StreamID[31:16]

Address bits[11:0] cannot be used.

The LHS can be one of: * PAS * MasterID/ExtendedID/UserFlags * Address

The RHS can be one of: * a numeric constant * SSD * T or negative version nT * StreamID

For realm (or 'Trusted') transactions, then SSD=0b11, T=1, nT=0. For non-secure (or 'Non-Trusted') transactions, then SSD=0b01, T=0, nT=1

Type

string

Default value

Address[43:12]=StreamID, PAS=SSD

percent_commit

Percentage of times that a read of a register with Update will commit the update. 0 means commit immediately.

Type

int

Default value

0x14

percent_commit_Update_clear

Percentage of times that a read of a register with a pending Update clear will lower the Update flag.

Type

int

Default value

0x14

pmu

What to instantiate as a PMU.

NOTE that all events and counters are intended for demonstration purposes only and should not be treated as in any way reflecting accurate values for a real implementation. The model's internal representation of actions differ significantly from real hardware and the particular value obtained from the counters should not be used for benchmarking.

Values of this parameter are: * "" -- no PMU * "distributed-0" * "distributed-1"

distributed-0: * a PMCG per TBU (number_of_ports, up to 63 ports) * a single PMCG for a TCU * Connect a debugger to see the configuration.

distributed-1: * same as distributed-0, except for supporting MSIs and MPAM on the MSIs if MPAM is supported by rest of the SMMU.

Type

string

Default value

ports_that_ignore_PnU_InD_on_transactions_with_no_SubstreamID

Some bus systems (notably PCIe) do not support marking a transaction as Privileged/User or Instruction/Data unless the transaction has a SubstreamID.

This accepts a comma separated list of numbers and ranges, for example:

0, 10-12, 15

If the number P is named in this list then the upstream pvbus_s[P] will have all transactions with no Substream considered to be User and Data.

Type

string

Default value**prefetch_only_requests**

The simulator supports 'prefetch-only' DMI requests, which can occur for anytime for any reason and are intended to be invisible to the end execution of the model and to the user.

0 -- deny all prefetch-only requests
 1 -- use debug requests for any page table walks -- form and use debug TLB/cache entries -- any faults will not record, but deny the prefetch request
 2 -- treat prefetch-only requests like normal transactions -- use normal page table walk transactions -- use and form normal TLB/cache entries -- faults will alter the programmer visible state of the SMMU

0 is the safest.

1 treats the access like a debug request and requires that debug page table walks are treated correctly downstream. Any descriptors that need HTTU to allow the transaction to proceed will fail the request.

2 is dangerous, it use real transactions and reports faults that are unphysical. Real transactions can be wait()ed and this disobeys the SystemC spec for get_direct_mem_ptr().

Type

int

Default value

0x0

ras

What to instantiate for RAS handling.

Values of this parameter are: * "" -- no RAS records * "MMU_600" * "MMU_700" * "MMU_S3"

MMU_600: * only corrected errors reported.

See also imp_def_ras_allow_non_secure_accesses_if_supports_secure.

Type

string

Default value**register_accesses_to_root_or_realm_pas_when_no_rme**

When RME is not implemented or disabled by legacy_tz_en: 0 -- root and realm register PAS accesses are not RAZ/WI 1 -- root and realm register PAS accesses are treated as RAZ/WI

Type

int

Default value

0x0

reset_value_of_SMMU_GBPA

Reset value of SMMU_GBPA

Type

int

Default value

0x0

reset_value_of_SMMU_S_GBPA

Reset value of SMMU_S_GBPA

Type

int

Default value

0x0

rme_ats_request_pa_strategy

When RME_IMPL == 0, the PA of an ATS Request's response is permitted but not required to undergo a GPT check:

0 -- do not check the PA 1 -- do the check against the PA 2 -- check the PA 50% of the time

Translated transactions are required to always undergo a GPT check whatever happens.

This parameter is ignored if RME_IMPL==1 and the PA is required to be checked.

Type

int

Default value

0x0

rme_da_force_better_configuration

RME-DA requires that the SMMU be integrated into a system for which SMMU_IDR0.COHA == 1 and SMMU_IDR0.IDR0.HTTU == both_af_and_dirty (2).

The model has pins:

* conf_system_supports_cohacc * conf_system_supports_httu

that can control these ID fields.

In addition, RME-DA requires that the fundamental SMMU has certain properties that are configured by its ID codes.

This parameter allows you to selectively ignore the pins and bad ID to produce a good configuration by forcing the required values.

Whether the SMMU has RME-DA or not is identified by SMMU_ROOT_IDR0.REALM_IMPL.

This is a comma-separated list of fields to force when RME-DA is configured by SMMU_ROOT_IDR0.REALM_IMPL:

* in SMMU_IDR0: * "Hyp" * "S1P" * "S2P" * "TTF" * "NS1ATS" * "COHACC" * "HTTU" * "RME_IMPL" * "SSIDSIZE" in SMMU_IDR1 * "BBML" in SMMU_IDR3

You can also use "all" to set all.

Type

string

Default value

rme_l0gpt_entry_covers_log2size_in_bytes

Each LOGPT entry covers: $2^{**rme_l0gpt_entry_covers_log2size_in_bytes}$ bytes of address space.

The valid values for this parameter are: * 30, 34, 36, 39

This parameter is reported in an encoded format as the read-only field: SMMU_ROOT_GPT_BASE_CFG.LOGPTSZ

This parameter can be overridden by the port l0gptsz_s when sampled on negedge of reset.

Type

int

Default value

0x1e

rme_speculation_control

This is a comma-separated list of flags that control when and how the model will perform speculation for RME.

Type

string

Default value

root_register_page_offset

This is the offset from SMMU_BASE of the Root register file page which is 64 KiB in size. It must not overlap any other part of the register map.

Type

int

Default value

0x0

secure_state_controls_access_to_SMMU_S_INIT

With RME access control of the SMMU_S_INIT belongs to Root. This parameters allows Root to delegate access control to the secure state, enabling secure software to reset the TLB, clearing out any TLB entries.

If RME is implemented and this parameter is 0, allow_non_secure_access_to_SMMU_S_INIT has no effect.

If the SMMU does not implement RME then this parameter is ignored.

Type

bool

Default value

0x1

seed

Used to seed the pseudo-random number generator that the SMMU model uses.

Type

int

Default value

0x12345678

separate_tw_msi_qs_port

True if there is a separate port which is used to walk configuration tables, translation tables, issue MSIs and access the queues. If this is false then pvbus_m[0] will be used.

Type

bool

Default value

0x1

size_of_cd_cache

The number of entries in the cache holding CD structures. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_dpttlb

The number of entries in the DPT TLB. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_gpttlb

The number of entries in the GPT TLB. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_llcd_cache

The number of entries in the cache holding L1CD descriptors. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_llste_cache

The number of entries in the cache holding L1STE descriptors. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_register_file

This is the power of two size that the register file occupies in the memory map. It is used to generate a mask for the addresses received on pbus_control_s to decode the desired register offset.

The default for this parameter is 1 MiB.

Type

int

Default value

0x100000

size_of_ste_cache

The number of entries in the cache holding STE structures. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

size_of_tlb

The number of entries in the TLB. If this is zero then it is treated as a large number ('infinite') but it is bounded so that the host memory usage of the cache is also bounded.

Type

int

Default value

0x0

smmu_msi_device_id

When appropriately enabled, assume that MSIs that are generated by the SMMU are presented to the GIC with this DeviceID.

See parameter msi_attribute_transform and enable_device_id_checks.

Type

int

Default value

0x0

smmu_v33_begin_offset_of_qcp0

This is the offset from SMMU_BASE of the first QCP page. The architecture requires that if more than one world of QCPs are present then they are in the order non-secure and then secure QCPs and form one continuous address space in the register file.

Type

int

Default value

0x0

support_for_httu_when_starts_disallowed

SMMU_IDR0.HTTU describes to the programmer whether the SMMU and system support HTTU. Typically an SMMU that is capable of HTTU will have a configuration pin that says whether the system supports HTTU or not.

The SMMU model determines SMMU_IDR0.HTTU as follows: * If the parameter SMMU_IDR0 indicates any kind of support for HTTU, then the configuration pin turns support on and off between that value and no support for HTTU. * If the parameter SMMU_IDR0 indicates no HTTU support, allow the pin to turn on support to that specified by this parameter.

Values for this parameter are the same as for the SMMU_IDR0.HTTU field: * 0 -- no support for HTTU * 1 -- AF flag only * 2 -- AF flag and DBM update

Type

int

Default value

0x0

tlb_when_do_f_tlb_conflict_on_overlap

If a TLB entry is created by a walk and it overlaps an existing entry. Then there are some architectural situations where the result is known. For all others, then an implementation is allowed to use an UNPREDICTABLE combination of the two entries, or it can generate F_TLB_CONFLICT:

0 -- never generate 1 -- sometimes generate 2 -- always generate

Conflicts between global and non-global entries are not detected by the model.

Type

int

Default value

0x0

translated_device_id_base

When appropriately enabled, assume that client device accesses are translated to a DeviceID as seen by the GIC of: StreamID + translated_device_id_base

See parameter output_attribute_transform and enable_device_id_checks.

Type

int

Default value

0x0

treat_debug_read_accesses_as_speculative_accesses

The SMMU architecture has the concept of speculative accesses. If you set this flag to true, then debug read accesses flowing from the upstream system through the SMMU will be interpreted as speculative.

The difference is that a speculative read will: * participate in HTTU * if it encounters a (non-HTTU) fault will always return abort

Debug writes are still considered as debug accesses. All speculative writes would be aborted and this is not a useful behaviour for the SMMU to emulate.

Type

bool

Default value

0x0

tw_qs_attribute_transform

Transform downstream attributes of table walk and queue transactions.

* ""/"none" -- no transform * How to alter the output attributes. Example:

"ExtendedID[35:32]=HWATTR_KIND_0"

RHS/LHS Symbols: * ExtendedID/MasterID/UserFlags

RHS Symbols: * HWATTR_KIND_0: PBHA information * kind * for a read: * 0/1 -- L1STE/STE * 2/3 -- L1CD/CD * 4/5 -- S1/S2 TTD (including CAS) * 6 -- CMDQ * 7 -- VMS * 11/12 -- LOGPT/L1GPT * 13/14 -- LODPT/L1DPT * for a write * 0 -- EVENTQ * 1 -- PRIQ * numeric literals.

ExtendedID/MasterID/UserFlags start with values {0, 0xFFFFFFFF, 0} respectively.

Any bits with no transform are unchanged.

NOTE: * see also 'output_attribute_transform' and 'msi_attribute_transform'.

Type

string

Default value**unpred_httu_percent_do_discretionary_AF**

If a descriptor could have a discretionary update of the AF flag on then what is the percentage of the time that the AF update should occur.

Type

int

Default value

0x32

unpred_httu_percent_do_discretionary_DBM

If a descriptor could have a discretionary DBM update to make the descriptor WriteableDirty then what is the percent of the time that the DBM update should occur.

Type

int

Default value

0x32

unpred_translated_access_out_of_range_of_oas

If a Translated Access is presented to the SMMU that is > OAS then it is CONSTRAINED UNPRED as to whether the transaction will either: 0 -- be truncated to OAS and go downstream 1 -- be aborted, no event written

Type

int

Default value

0x1

wait_atos_ticks

This is the time to wait before doing an ATOS operation. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_cmdq_ticks

This is the time to wait before doing something on the command queue. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_eventq_ticks

This is the time to wait before doing something on the event queue. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_misc_async_actions_ticks

This is the time to wait before doing an async action that could be delayed is performed. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_msi_ticks

This is the time to wait before sending an MSI. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_pri_req_ticks

This is the time to wait before processing a PRI Request. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x0

wait_pri_resp_ticks

This is the time to wait before sending a PRI Response back to the PCIe subsystem. When a PRI Response is an auto-response then the ATC might immediately make a new ATS request, that immediately fails, that immediately makes a PRI Request, that auto-responds, etc. To break this loop, then we introduce a minimum time on all PRI Responses to give other components in the system a chance to run. If bit 32 is set (0x1_0000_0000) then the time waited for is a uniform randomly distributed time [0,max(2,(t & 0xFFFFffff))).

Type

int

Default value

0x1

when_fetch_vms
Architecturally, there is flexibility in how a VMS is cached and thus: * when it will be fetched
* the prioritization of F_VMS_FETCH.

Of the many architecturally-allowed options, the model offers two:

0 -- fetched and cached immediately after the STE is fetched
1 -- fetched and cached immediately after the CD is fetched

In both cases, then the VMS is cached in the STE and CMD_CFGI_VMS_PIDM is a NOP.

Type
int
Default value
0x0

width_of_agbpa_impdef
Width of the SMMU_s_AGBPA.IMPDEF field.
Type
int
Default value
0x10

3.10.87 SMSC_91C111

10/100 Non-PCI Ethernet Controller(SMSC 91C111). This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1180: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for SMSC_91C111

This model has the following Iris instances:

Table 3-1181: SMSC_91C111 Iris instances

InstanceName	ComponentName
SMSC_91C111	SMSC_91C111
SMSC_91C111.SMSC_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-1182: SMSC_91C111 MTI instances

InstanceName	ComponentName
SMSC_91C111.SMSC_slave	PVBusSlave

SMSC_91C111 contains the following CADI targets:

- SMSC_91C111

About SMSC_91C111

This component provides the register interface of the SMSC part and can be configured to act as an unconnected Ethernet port, or an Ethernet port connected to the host by an Ethernet bridge.

It uses a banked register model of primarily 16-bit registers. There are also indirectly accessible registers for the PHY unit.

If a MAC address is not specified in the `mac_address` parameter, the simulator takes the default MAC address, which is randomly generated. This provides some degree of MAC address uniqueness when running models on multiple hosts on a local network.



DHCP servers allocate the IP addresses, but because they sometimes do this based on the MAC address provided to them, using random MAC addresses might interact unfortunately with some DHCP servers.

See also:

- [1.11.3 Configuring the networking environment for Microsoft Windows](#) on page 47
- [1.11.8 Configuring the networking environment for Linux](#) on page 53

Ports for SMSC_91C111

Table 3-1183: Ports

Name	Protocol	Type	Description
clock	ClockSignal	Slave	Clock input, typically 25MHz, which sets the master transmit/receive rate.
eth	VirtualEthernet	Master	Ethernet port.
intr	Signal	Master	Interrupt signal.
pvbus	PVBus	Slave	Slave port for register access.
state	ValueState_64	Master	State port to retrieve state of host bridge

Parameters for SMSC_91C111

cache_size

Size of cache memory in SMSC MMU

Type

int

Default value

0x10000

enabled

Host interface connection enabled

Type

bool

Default value

0x0

mac_address

Host/model MAC address

Type

string

Default value

00:02:f7:ef:5f:50

not_lan911x

Gracefully fail SMSC LAN911x driver probe

Type

bool

Default value

0x0

promiscuous

Put host into promiscuous mode

Type

bool

Default value

0x1

3.10.88 SP804_Timer

ARM Dual-Timer Module(SP804). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1184: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for SP804_Timer

This model has the following Iris instances:

Table 3-1185: SP804_Timer Iris instances

InstanceName	ComponentName
SP804_Timer	SP804_Timer
SP804_Timer.busslave	PVBusSlave
SP804_Timer.clk_div0	ClockDivider
SP804_Timer.clk_div1	ClockDivider
SP804_Timer.counter0	CounterModule
SP804_Timer.counter1	CounterModule

This model has the following MTI trace components:

Table 3-1186: SP804_Timer MTI instances

InstanceName	ComponentName
SP804_Timer.busslave	PVBusSlave
SP804_Timer.clk_div0	ClockDivider
SP804_Timer.clk_div1	ClockDivider

SP804_Timer contains the following CADI targets:

- ClockDivider
- CounterModule
- SP804_Timer

Ports for SP804_Timer

Table 3-1187: Ports

Name	Protocol	Type	Description
clock	ClockSignal	Slave	Clock input, typically 1MHz, driving master count rate.
irq_out0	Signal	Master	Interrupt signaling.
irq_out1	Signal	Master	Interrupt signaling.
pvbus	PVBus	Slave	Slave port for register access.
timer_en[2]	ClockRateControl	Slave	Port for changing the rate of timer n.

Parameters for SP804_Timer

clk_div0.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clk_div0.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type
int

Default value
0x1

clk_div1.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type
int

Default value
0x1

clk_div1.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type
int

Default value
0x1

3.10.89 SP805_Watchdog

ARM Watchdog Module(SP805). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1188: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for SP805_Watchdog

This model has the following Iris instances:

Table 3-1189: SP805_Watchdog Iris instances

InstanceName	ComponentName
SP805_Watchdog	SP805_Watchdog
SP805_Watchdog.busslave	PVBusSlave
SP805_Watchdog.clocktimer	ClockTimerThread
SP805_Watchdog.clocktimer.timer	ClockTimerThread64

InstanceName	ComponentName
SP805_Watchdog.clocktimer.timer.thread	SchedulerThread
SP805_Watchdog.clocktimer.timer.thread_event	SchedulerThreadEvent

This model has the following MTI trace components:

Table 3-1190: SP805_Watchdog MTI instances

InstanceName	ComponentName
SP805_Watchdog.busslave	PVBusSlave

SP805_Watchdog contains the following CADI targets:

- ClockTimerThread
- ClockTimerThread64
- SP805_Watchdog
- SchedulerThread
- SchedulerThreadEvent

Ports for SP805_Watchdog

Table 3-1191: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Clock input, typically 1MHz, driving master count rate.
irq_out	Signal	Master	Interrupt signaling.
pvbus_s	PVBus	Slave	Slave port for register access.
reset_in	Signal	Slave	Reset signaling.
reset_out	Signal	Master	Reset signaling.

Parameters for SP805_Watchdog

simhalt

Halt on reset.

Type

bool

Default value

0x0

3.10.90 SP810_SysCtrl

PrimeXsys System Controller(SP810) NB: Only EB relevant functionalities are fully implemented. This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1192: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for SP810_SysCtrl

This model has the following Iris instances:

Table 3-1193: SP810_SysCtrl Iris instances

InstanceName	ComponentName
SP810_SysCtrl	SP810_SysCtrl
SP810_SysCtrl.busslave	PVBusSlave
SP810_SysCtrl.clkdiv_clk0	ClockDivider
SP810_SysCtrl.clkdiv_clk1	ClockDivider
SP810_SysCtrl.clkdiv_clk2	ClockDivider
SP810_SysCtrl.clkdiv_clk3	ClockDivider

This model has the following MTI trace components:

Table 3-1194: SP810_SysCtrl MTI instances

InstanceName	ComponentName
SP810_SysCtrl.busslave	PVBusSlave
SP810_SysCtrl.clkdiv_clk0	ClockDivider
SP810_SysCtrl.clkdiv_clk1	ClockDivider
SP810_SysCtrl.clkdiv_clk2	ClockDivider
SP810_SysCtrl.clkdiv_clk3	ClockDivider

SP810_SysCtrl contains the following CADI targets:

- ClockDivider
- SP810_SysCtrl

Ports for SP810_SysCtrl

Table 3-1195: Ports

Name	Protocol	Type	Description
clk_in	ClockSignal	Slave	Clock input.
hclkdivsel	ValueState	Master	Define the processor clock/bus clock ratio. Not fully implemented. Using this port has unpredictable results.
npwr	Signal	Slave	Power on reset. Not fully implemented. Using this port has unpredictable results.
pll_en	Signal	Master	PLL enable output. Not fully implemented. Using this port has unpredictable results.
pvbus	PVBus	Slave	Slave port for register access.
ref_clk_in	ClockSignal	Slave	Clock source used by the Timer and Watchdog modules.

Name	Protocol	Type	Description
remap_clear	StateSignal	Master	Remap clear request output.
remap_stat	StateSignal	Slave	Remap status input. Not fully implemented. Using this port has unpredictable results.
sleep_mode	Signal	Master	Control clocks for SLEEP mode. Not fully implemented. Using this port has unpredictable results.
sys_id	ValueState	Slave	Unused port.
sys_mode	ValueState	Slave	Present system mode. Not fully implemented. Using this port has unpredictable results.
sys_stat	ValueState	Slave	System status input. Not fully implemented. Using this port has unpredictable results.
timer_clk_en[4]	ClockRateControl	Master	Timer clock enable n.
wd_clk_en	Signal	Master	Watchdog module clock enable output. Not fully implemented. Using this port has unpredictable results.
wd_en	Signal	Slave	Watchdog module enable input. Not fully implemented. Using this port has unpredictable results.

Parameters for SP810_SysCtrl

clkdiv_clk0.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv_clk0.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv_clk1.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv_clk1.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv_clk2.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv_clk2.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv_clk3.div

Clock Rate Divider. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

clkdiv_clk3.mul

Clock Rate Multiplier. This parameter is not exposed via CADI and can only be set in LISA

Type

int

Default value

0x1

sysid

System Identification Register.

Type

int

Default value

0x0

use_s8

Use Switch 8 (S1-S4)

Type

bool

Default value

0x0

3.10.91 SSU

Safety Status Unit. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1196: IP revisions support

Revision	Quality level
r0p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for SSU

This model has the following Iris instances:

Table 3-1197: SSU Iris instances

InstanceName	ComponentName
SSU	SSU
SSU.pvbus_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-1198: SSU MTI instances

InstanceName	ComponentName
SSU	SSU
SSU.pvbus_slave	PVBusSlave

SSU contains the following CADI targets:

- SSU

Ports for SSU

Table 3-1199: Ports

Name	Protocol	Type	Description
c_error_in	Signal	Slave	Critical interrupts to the SSU
clk_in	ClockSignal	Slave	Clock input

Name	Protocol	Type	Description
cold_reset_in	Signal	Slave	Cold reset signal
nc_error_in	Signal	Slave	Non critical interrupts to the SSU
pvbuss_s	PVBus	Slave	To access FMU model registers
ssu_out	ValueState	Master	SSU output port
warm_reset_in	Signal	Slave	Warm reset signal

Parameters for SSU

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT:2)

Type

int

Default value

0x2

sm_implemented

Safety Mechanism Implemented

Type

bool

Default value

0x0

3.10.92 SecureAlarmManager

Security Alarm Manager. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1200: IP revisions support

Revision	Quality level
0.93	Alpha support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for SecureAlarmManager

This model has the following Iris instances:

Table 3-1201: SecureAlarmManager Iris instances

InstanceName	ComponentName
SecureAlarmManager	SecureAlarmManager
SecureAlarmManager.apb	PVBusSlave

This model has the following MTI trace components:

Table 3-1202: SecureAlarmManager MTI instances

InstanceName	ComponentName
SecureAlarmManager.apb	PVBusSlave

SecureAlarmManager contains the following CADI targets:

- SecureAlarmManager

Ports for SecureAlarmManager

Table 3-1203: Ports

Name	Protocol	Type	Description
apb	PVBus	Slave	APB Subordinate Interface - Access to registers
clk_in	ClockSignal	Slave	Clock in signal
config_done_trig_ack_in	Signal	Slave	Config done ack signal
config_done_trig_req_out	Signal	Master	Config done req signal
event_in[61]	Signal	Slave	Event in signal
event_status_out[64]	Signal	Master	Event status out signal
nCOLDRESETAON_in	Signal	Slave	Coldreset in signal
reset_in	Signal	Slave	Reset in signal
response_action_out[8]	Signal	Master	Response action out signal

Parameters for SecureAlarmManager

NUM_SAMNEC

Number of SAM event counters

Type

int

Default value

0x3

NUM_SAMNRA

Number of SAM response actions

Type

int

Default value

0x7

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type

int

Default value

0x2

3.10.93 SecureICache

SecureICache. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1204: IP revisions support

Revision	Quality level
1.13	Alpha support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Changes in 11.26.8

Parameters added:

- SIC_MAX_CODE_SIZE

Iris and MTI instances for SecureICache

This model has the following Iris instances:

Table 3-1205: SecureICache Iris instances

InstanceName	ComponentName
SecureICache	SecureICache
SecureICache.DECRYPT_RAM_0	PVBusSlave
SecureICache.DECRYPT_RAM_1	PVBusSlave
SecureICache.HTR_RAM	PVBusSlave
SecureICache.SIC_BusMapper	PVBusMapper
SecureICache.SIC_BusMapper0	PVBusMapper
SecureICache.SIC_BusMapper1	PVBusMapper
SecureICache.ZERO_RAM	PVBusSlave
SecureICache.apb	PVBusSlave

This model has the following MTI trace components:

Table 3-1206: SecureICache MTI instances

InstanceName	ComponentName
SecureICache.DECRYPT_RAM_0	PVBusSlave
SecureICache.DECRYPT_RAM_1	PVBusSlave
SecureICache.HTR_RAM	PVBusSlave
SecureICache.SIC_BusMapper	PVBusMapper

InstanceName	ComponentName
SecureICache.SIC_BusMapper0	PVBusMapper
SecureICache.SIC_BusMapper1	PVBusMapper
SecureICache.ZERO_RAM	PVBusSlave
SecureICache.apb	PVBusSlave

SecureICache contains the following CADI targets:

- SecureICache

Ports for SecureICache

Table 3-1207: Ports

Name	Protocol	Type	Description
apb	PVBus	Slave	APB4 Subordinate Interface - Access to registers
htr_ram_pvbus_s	PVBus	Slave	To Read and Write pre-calculated SHA-256 Digests
irq_out	Signal	Master	To indicate interrupts
pvbus_m	PVBus	Master	To read from External RAM
pvbus_s	PVBus	Slave	Receives Transactions from CPU
reset_in	Signal	Slave	reset

Parameters for SecureICache

SIC_AUTH_ENABLE

SIC Authentication enabled [default==false]

Type

bool

Default value

0x0

SIC_DECRYPT_ENABLE

SIC Decryption enabled [default==false]

Type

bool

Default value

0x0

SIC_DR_CNT

SIC config: Decryption Region Count (0x0==1, 0x1==2[default], 0x2==4)

Type

int

Default value

0x1

SIC_HTR_RAM_SIZE

SIC config: Hash Tag RAM Size (0x1==1KB, 0x2==2KB, 0x4==4KB, 0x8==8KB, 0x10=16KB, 0x20=32KB[default])

Type

int

Default value

0x20

SIC_MAX_CODE_SIZE

SIC config: Maximum Code Size supported (n==nKB[default,1024(1MB)])

Type

int

Default value

0x400

SIC_PAGE_RAM_SIZE

SIC config: Page RAM Size (0x4==4KB, 0x8==8KB[default], 0x10=16KB)

Type

int

Default value

0x10

SIC_PAGE_SIZE

SIC config: Page Size (0x0==128B, 0x1==256B, 0x2=512B, 0x3=1KB[default], 0x4=2KB, 0x5=4KB)

Type

int

Default value

0x3

SIC_PMON_EN

SIC config: Performance Montior enable [default==false]

Type

bool

Default value

0x0

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type

int

Default value

0x2

3.10.94 SystemFMU

System level Fault Management Unit. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1208: IP revisions support

Revision	Quality level
rOp0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for SystemFMU

This model has the following Iris instances:

Table 3-1209: SystemFMU Iris instances

InstanceName	ComponentName
SystemFMU	SYSTEM_FMU
SystemFMU.pvbus_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-1210: SystemFMU MTI instances

InstanceName	ComponentName
SystemFMU	SYSTEM_FMU
SystemFMU.pvbus_slave	PVBusSlave

SystemFMU contains the following CADI targets:

- SYSTEM_FMU

Ports for SystemFMU

Table 3-1211: Ports

Name	Protocol	Type	Description
c_error_in[26]	Signal	Slave	Critical interrupts from the device side FMUs
c_error_out	Signal	Master	Critical error input from the system FMU
clk_in	ClockSignal	Slave	Clock input
cold_reset_in	Signal	Slave	Cold reset signal
nc_error_in[26]	Signal	Slave	Non critical interrupts from the device side FMUs
nc_error_out	Signal	Master	Non critical error input from the system FMU
pvbus_s	PVBus	Slave	To access FMU model registers

Name	Protocol	Type	Description
warm_reset_in	Signal	Slave	Warm reset signal

Parameters for SystemFMU

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type

int

Default value

0x2

upstream_fmus_cfg

Number of upstream FMUs that can be connected to the System level FMU

Type

int

Default value

0x1

3.10.95 System_RAS_Agent

System level RAS agent. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1212: IP revisions support

Revision	Quality level
rOp0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for System_RAS_Agent

This model has the following Iris instances:

Table 3-1213: System_RAS_Agent Iris instances

InstanceName	ComponentName
System_RAS_Agent	SYSTEM_RAS_AGENT
System_RAS_Agent.PVBusSubordinate	PVBusSlave

This model has the following MTI trace components:

Table 3-1214: System_RAS_Agent MTI instances

InstanceName	ComponentName
System_RAS_Agent.PVBusSubordinate	PVBusSlave

System_RAS_Agent contains the following CADI targets:

- SYSTEM_RAS_AGENT

Ports for System_RAS_Agent

Table 3-1215: Ports

Name	Protocol	Type	Description
apb	PVBus	Slave	To access RAS agent registers
clk_in	ClockSignal	Slave	Input Clock - RAS agent is in this clock domain
cri_in[56]	Signal	Slave	Critical Error Interrupt from the downstream RAS agent
cri_out	Signal	Master	CRI_OUT is the consolidated status of the Critical Error Interrupt from this and the downstream RAS agent(s).
eri_in[56]	Signal	Slave	Error Recovery Interrupt from the downstream RAS agent
eri_out	Signal	Master	consolidated status of the Error Recovery Interrupt from this and the downstream RAS agent(s).
fhi_in[56]	Signal	Slave	Fault Handling Interrupt from the downstream RAS agent
fhi_out	Signal	Master	consolidated status of the Fault Handling Interrupt from this and the downstream RAS agent(s).
reset_in	Signal	Slave	Input reset - Connect to the system cold reset.
valid_in[56]	Signal	Slave	Incoming Valid from a downstream RAS agent indicates the presence of at least one valid error record.
valid_out	Signal	Master	outgoing Valid from the downstream RAS agent(s) contains at least one valid error record.

Parameters for System_RAS_Agent

NUM_DOWNSTREAM_RAS_AGENTS

Number of downstream RAS agents for which the proxy error record is maintained

Type

int

Default value

0x1

SYNC_ENABLE

Enables synchronization on the Interrupt lines and valid line before assigning it to the corresponding bits in the ERR<n>STATUS register. 1-bit per downstream RAS Agent.

Type

int

Default value

0x0

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type

int

Default value

0x2

3.10.96 TZC_400

TrustZone Address Space Controller. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1216: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for TZC_400

This model has the following Iris instances:

Table 3-1217: TZC_400 Iris instances

InstanceName	ComponentName
TZC_400	TZC_400
TZC_400.apbslave[0]	PVBusSlave
TZC_400.filter0	filter0
TZC_400.filter0.BusMapper	PVBusMapper
TZC_400.filter1	filter1
TZC_400.filter1.BusMapper	PVBusMapper
TZC_400.filter2	filter2
TZC_400.filter2.BusMapper	PVBusMapper
TZC_400.filter3	filter3
TZC_400.filter3.BusMapper	PVBusMapper

This model has the following MTI trace components:

Table 3-1218: TZC_400 MTI instances

InstanceName	ComponentName
TZC_400.apbslave[0]	PVBusSlave
TZC_400.filter0.BusMapper	PVBusMapper
TZC_400.filter1.BusMapper	PVBusMapper

InstanceName	ComponentName
TZC_400.filter2.BusMapper	PVBusMapper
TZC_400.filter3.BusMapper	PVBusMapper

TZC_400 contains the following CADI targets:

- TZC_400

About TZC_400

The TZC-400 determines, under software control, whether a particular bus master is permitted to issue Non-secure accesses to a particular physical address.

The component has:

- Eight address regions in addition to the base region, region 0.
- A programmable control block for security-access permissions configuration through the *Advanced Peripheral Bus* (APB).
- Up to four address filters that share common set region set-up registers.
- Software configurable permission check failure reporting and interrupt signaling.
- Filtering with a *Non-Secure Access ID* (NSAID).
- A gate keeper, to allow or block accesses to the filter unit.
- Configurable reset values of region configuration registers and other key configuration registers.

This component has the following subcomponents:

TZFilterUnits

The TZC-400 has four TZFilterUnits. The `BUILD_CONFIG` register sets the configuration. The `rst_build_config` parameter controls the register. The value of `rst_build_config` varies with the system. See the system design documentation or system integration documentation. For AEMvA, it is `0x3003F08`.

TZDummyDevice

An internal dummy device that mimics RAZ/WI for TZFilterUnits. The system uses it when there is a permission violation and a bus returns Transaction OK.



Note

- Configure `master_id_from_label` OR `id_mapping`, `rst_build_config`, and `rst_region_attributes_0` before running the model to set the desired behaviors. Otherwise, the system resets all region configuration registers, `rst_action`, and `rst_gate_keeper` to 0, and resets `rst_build_config` and `rst_region_attributes_0` to sensible default values.
- Configure either `id_mapping` OR `master_id_from_label` at model init, or a warning message appears.
- The syntax of `id_mapping` is:

```
<masterid_0>:<nsaid_0>,<masterid_1>:<nsaid_1>,<masterid_n>:<nsaid_n>
```

Separate the mapping pairs by a comma. The `masterid` is the ID of the bus master, such as the parameter `CLUSTER_ID` on Cortex-A15/7, `cluster_id` port of Cortex-A15/7, or `master_id` parameter for Cortex-M3.

Differences between the model and the RTL

Unlike the hardware, this component does not have:

- Asynchronous clocks. The model does not need clocks for data transfer, or clock signals.
- QoS Virtual Network (QVN) support. Specifically, it does not implement the `vnet` bits[27:24] in `FAIL_ID_<x>` registers.
- Fast Path and Fast Path ID. In the model, transactions occur at similar speeds.
- 256 outstanding accesses globally for each read or write Normal Paths and configurable 8, 16, or 32 outstanding accesses on Fast Path read access. The model does not support QVN. This concept is meaningless for a PV level model.
- Configurable address bus width, data bus width, transaction ID tag, and USER bus width. A single bus implementation, PVBUS, covers these AXI bus hardware implementation details.

This component does not implement:

- The `vnet` bits[27:24] in `FAIL_ID_<x>` registers.
- Any background logic for the speculation control register. This does not affect model behavior.

Ports for TZC_400

Table 3-1219: Ports

Name	Protocol	Type	Description
<code>apbslave_s</code>	PVBUS	Slave	Bus access for control register.
<code>filter_pvbus_m[4]</code>	PVBUS	Master	Outgoing bus traffic from filter units.
<code>filter_pvbus_s[4]</code>	PVBUS	Slave	Incoming bus traffic to filter units.
<code>tzc_reset</code>	Signal	Slave	Reset signal from external master.
<code>tzcint</code>	Signal	Master	TrustZone interrupt signal, controlled by ACTION register.

Parameters for TZC_400

diagnostics

Diagnostics

Type

int

Default value

0x0

id_mapping

Mapping from Master ID to NSAID

Type

string

Default value

0:0,1:0,2:0,3:0,4:0,5:0,6:0,7:0,8:0,9:0,10:0,11:0,12:0,13:0,14:0,15:0

master_id_from_label

Obtain Master ID from label (ignores id_mapping)

Type

bool

Default value

0x0

rst_action

ACTION register value at reset

Type

int

Default value

0x0

rst_build_config

BUILD_CONFIG register value at reset

Type

int

Default value

0x3003f08

rst_gate_keeper

GATE_KEEPER register value at reset

Type

int

Default value

0x0

rst_region_attributes_0

Region 0 Secure attributes

Type

int

Default value

0xf

rst_region_attributes_1

Region 1 Secure attributes

Type

int

Default value

0x0

rst_region_attributes_2

Region 2 Secure attributes

Type

int

Default value

0x0

rst_region_attributes_3

Region 3 Secure attributes

Type

int

Default value

0x0

rst_region_attributes_4

Region 4 Secure attributes

Type

int

Default value

0x0

rst_region_attributes_5

Region 5 Secure attributes

Type

int

Default value

0x0

rst_region_attributes_6

Region 6 Secure attributes

Type

int

Default value

0x0

rst_region_attributes_7

Region 7 Secure attributes

Type

int

Default value

0x0

rst_region_attributes_8

Region 8 Secure attributes

Type

int

Default value

0x0

rst_region_base_high_1

Region 1 base memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_base_high_2

Region 2 base memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_base_high_3

Region 3 base memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_base_high_4

Region 4 base memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_base_high_5

Region 5 base memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_base_high_6

Region 6 base memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_base_high_7

Region 7 base memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_base_high_8

Region 8 base memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_base_low_1

Region 1 base memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_base_low_2

Region 2 base memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_base_low_3

Region 3 base memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_base_low_4

Region 4 base memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_base_low_5

Region 5 base memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_base_low_6

Region 6 base memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_base_low_7

Region 7 base memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_base_low_8

Region 8 base memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_id_access_0

Region 0 NSAID permissions

Type

int

Default value

0x0

rst_region_id_access_1

Region 1 NSAID permissions

Type

int

Default value

0x0

rst_region_id_access_2

Region 2 NSAID permissions

Type

int

Default value

0x0

rst_region_id_access_3

Region 3 NSAID permissions

Type

int

Default value

0x0

rst_region_id_access_4

Region 4 NSAID permissions

Type

int

Default value

0x0

rst_region_id_access_5

Region 5 NSAID permissions

Type

int

Default value

0x0

rst_region_id_access_6

Region 6 NSAID permissions

Type

int

Default value

0x0

rst_region_id_access_7

Region 7 NSAID permissions

Type

int

Default value

0x0

rst_region_id_access_8

Region 8 NSAID permissions

Type

int

Default value

0x0

rst_region_top_high_0

Region 0 (default) top memory address

Type

int

Default value

0x0

rst_region_top_high_1

Region 1 top memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_top_high_2

Region 2 top memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_top_high_3

Region 3 top memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_top_high_4

Region 4 top memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_top_high_5

Region 5 top memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_top_high_6

Region 6 top memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_top_high_7

Region 7 top memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_top_high_8

Region 8 top memory address (high 32 bits)

Type

int

Default value

0x0

rst_region_top_low_1

Region 1 top memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_top_low_2

Region 2 top memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_top_low_3

Region 3 top memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_top_low_4

Region 4 top memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_top_low_5

Region 5 top memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_top_low_6

Region 6 top memory address (low 32 bits)

Type

int

Default value

0x0

rst_region_top_low_7
Region 7 top memory address (low 32 bits)

Type
int

Default value
0x0

rst_region_top_low_8
Region 8 top memory address (low 32 bits)

Type
int

Default value
0x0

3.10.97 TZFilterUnit

TrustZone Filter Unit. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1220: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for TZFilterUnit

This model has the following Iris instances:

Table 3-1221: TZFilterUnit Iris instances

InstanceName	ComponentName
TZFilterUnit	TZFilterUnit
TZFilterUnit.BusMapper	PVBusMapper

This model has the following MTI trace components:

Table 3-1222: TZFilterUnit MTI instances

InstanceName	ComponentName
TZFilterUnit.BusMapper	PVBusMapper

TZFilterUnit contains the following CADI targets:

- TZFilterUnit

Ports for TZFilterUnit

Table 3-1223: Ports

Name	Protocol	Type	Description
control	TZFilterControl	Master	Configuration port.
pdbus_m	PVBus	Master	Master bus port.
pdbus_s	PVBus	Slave	Slave bus port.

3.10.98 TZIC

ARM TrustZone Interrupt Controller(SP890). This model is written in LISA+.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1224: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for TZIC

This model has the following Iris instances:

Table 3-1225: TZIC Iris instances

InstanceName	ComponentName
TZIC	TZIC
TZIC.busslave	PVBusSlave

This model has the following MTI trace components:

Table 3-1226: TZIC MTI instances

InstanceName	ComponentName
TZIC.busslave	PVBusSlave

TZIC contains the following CADI targets:

- [TZIC](#)

About TZIC

The TZIC provides a software interface to the secure interrupt system in a TrustZone® design. It provides secure control of the nFIQ and masks out the interrupt sources chosen for nFIQ from the interrupts that are passed onto a non-secure interrupt controller.

Ports for TZIC

Table 3-1227: Ports

Name	Protocol	Type	Description
fiq_out	Signal	Master	FIQ interrupt to processor.
input[32]	Signal	Slave	32 interrupt input sources.
irq_out[32]	Signal	Master	32 IRQ output ports.
nsfiq_in	Signal	Slave	Connects to the nFIQ output of the non-secure interrupt controller.
pvbuss	PVBus	Slave	Slave port for connection to PV bus master/decoder.
sfiq_in	Signal	Slave	Daisy chaining secure FIQ input, otherwise connects to logic 1 if interrupt controller not daisy chained.

3.10.99 TrustedRAM

Trusted RAM. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1228: IP revisions support

Revision	Quality level
0.4	Alpha support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for TrustedRAM

This model has the following Iris instances:

Table 3-1229: TrustedRAM Iris instances

InstanceName	ComponentName
TrustedRAM	TrustedRAM
TrustedRAM.apb	PVBusSlave

This model has the following MTI trace components:

Table 3-1230: TrustedRAM MTI instances

InstanceName	ComponentName
TrustedRAM.apb	PVBusSlave

TrustedRAM contains the following CADI targets:

- TrustedRAM

Ports for TrustedRAM

Table 3-1231: Ports

Name	Protocol	Type	Description
apb	PVBus	Slave	APB Subordinate Interface - Access to registers
reset_in	Signal	Slave	Reset in signal

Parameters for TrustedRAM

TRBC_RESET_VALUE

TRBC Registers Reset Value

Type

int

Default value

0xb

diagnostics

Diagnostics 0-4 (0:FATAL 1:ERROR 2:WARNING 3:INFO 4:DEBUG - DEFAULT==2)

Type

int

Default value

0x2

3.10.100 VHT_VIOBridge

Arm VHT Virtual I/O interface. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1232: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Ports for VHT_VIOBridge

Table 3-1233: Ports

Name	Protocol	Type	Description
pvbuss	PVBus	Slave	Target port for access to VIOBridge registers

Parameters for VHT_VIOBridge

vio_basename

Basename of scripts to use for VIO

Type

string

Default value

arm_vio

vio_path

Path to find python scripts for VIO

Type

string

Default value

3.10.101 VHT_VSIBridge

Arm VHT Virtual stream interface. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1234: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Ports for VHT_VSIBridge

Table 3-1235: Ports

Name	Protocol	Type	Description
intr	Signal	Master	Interrupt raising signal
pvbus_m	PVBus	Master	Requester port for access to emeory
pvbus_s	PVBus	Slave	Completer port for access to VSIBridge registers

Parameters for VHT_VSIBridge

vsi_basename

Basename of scripts to use for VSI

Type

string

Default value

arm_vsi

vsi_idx

Index to use for VSI

Type

int

Default value

0x0

vsi_path

Path to find python scripts for VSI

Type

string

Default value

3.10.102 VHT_VSocket

Arm VHT Virtual socket bridge interface. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1236: IP revisions support

Revision	Quality level
r1p0	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Ports for VHT_VSocket

Table 3-1237: Ports

Name	Protocol	Type	Description
pvbus_m	PVBus	Master	Port for VSocket bridge to access external memory
pvbus_s	PVBus	Slave	Target port for access to VSocket bridge registers

Parameters for VHT_VSocket

name

Name of the component

Type

string

Default value

3.10.103 v7_VGIC

System VGIC architecture version v7. This model is written in C++.

This model supports the following revisions of the IP at the given quality levels:

Table 3-1238: IP revisions support

Revision	Quality level
N/A	Full support

For an explanation of the quality levels, see [1.3 Quality level definitions](#) on page 22.

Iris and MTI instances for v7_VGIC

This model has the following Iris instances:

Table 3-1239: v7_VGIC Iris instances

InstanceName	ComponentName
v7_VGIC	v7_VGIC
v7_VGIC.vgic_bus_slave	PVBusSlave

This model has the following MTI trace components:

Table 3-1240: v7_VGIC MTI instances

InstanceName	ComponentName
v7_VGIC	v7_VGIC
v7_VGIC.vgic_bus_slave	PVBusSlave

v7_VGIC contains the following CADI targets:

- v7_VGIC

Ports for v7_VGIC

Table 3-1241: Ports

Name	Protocol	Type	Description
cfgsdisable	Signal	Slave	Disable write access to some GIC registers.
configuration	v7_VGIC_Configuration_Protocol	Slave	Configure the mapping of the core number (from MasterID) to the core interface number.
fiq_in[8]	Signal	Slave	FIQ inputs.
fiq_out[8]	Signal	Master	FIQ outputs.
irq_in[8]	Signal	Slave	IRQ inputs.
irq_out[8]	Signal	Master	IRQ outputs.
ppi_core0[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu 0.
ppi_core1[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu 1.
ppi_core2[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu 2.
ppi_core3[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu 3.
ppi_core4[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu 4.
ppi_core5[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu 5.
ppi_core6[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu 6.
ppi_core7[16]	Signal	Slave	Private peripheral interrupts (ID16-ID31) for cpu 7.

Name	Protocol	Type	Description
pvbus_s	PVBus	Slave	Bus port for accessing distributor registers.
reporting_interface	VGICReportingProtocol	Slave	Logging interface.
reset_signal	Signal	Slave	Reset signal input.
spi[988]	Signal	Slave	SPI inputs.
vfiq_out[8]	Signal	Master	Virtual FIQ outputs.
virq_out[8]	Signal	Master	Virtual IRQ outputs.
wakeup_fiq[8]	Signal	Master	Wakeup signal for FIQ.
wakeup_irq[8]	Signal	Master	Wakeup signal for IRQ.

Parameters for v7_VGIC

core-impl-id

Implementation ID to present for the cores

Type

int

Default value

0x3902043b

dist-impl-id

Implementation ID to present for the distributor

Type

int

Default value

0x3902043b

enable_log_errors

Type

bool

Default value

0x0

enable_log_fatal

Type

bool

Default value

0x0

enable_log_warnings

Type

bool

Default value

0x0

enabled

Enable the component. If it is disabled then all register writes will have no effect.

Type

bool

Default value

0x1

number-of-cores

Number of core interfaces to present

Type

int

Default value

0x8

number-of-ints

Number of interrupt pins. Will be rounded down to the nearest multiple of 32

Type

int

Default value

0xe0

number-of-lrs

Number of list registers

Type

int

Default value

0x40

vgic-version

Version number of the VGIC interface

Type

int

Default value

0xb

4. Plug-ins for Fast Models

This chapter describes the plug-ins that are available for Fast Models.

Prebuilt plug-ins can be found at `$PVLIB_HOME/plugins/<OS_compiler>/`. The source code for some of these plug-ins is provided as programming examples, under `$PVLIB_HOME/examples/MTI/`.

4.1 Loading a plug-in

The method of loading a plug-in depends on the type of model being used.

4.1.1 --plugin command-line option

To load a plug-in, use the `--plugin <path_to_plugin>/<plugin_name>` option.

To load more than one plug-in, use multiple `--plugin` options.

Specify plug-in parameters using one or more `-c` options when launching the model, or use a configuration file.

For example the following command:

- Loads the TarmacTrace plug-in.
- Sets the TarmacTrace `trace-file` parameter.

```
./isim_system --plugin $plugin_dir/TarmacTrace.so -C TRACE.TarmacTrace.trace-  
file=tTrace.log
```

Related information

- [scx::scx_parse_and_configure](#)

4.1.2 scx::scx_load_plugin() method

Fast Models that are exported to SystemC can call the method `scx::scx_load_plugin()` to hard code the path to the plug-in, before calling `sc_start()`.

For example:

```
scx::scx_load_plugin("$PVLIB_HOME/plugins/Linux64_GCC-10.3/TarmacTrace.so");
```

Related information

- [scx::scx_load_plugin](#)

4.1.3 FM_TRACE_PLUGINS environment variable

If it is not possible to specify a trace plug-in to the launching tool, use the environment variable `FM_TRACE_PLUGINS`. This variable must be set to the full path of the plug-in.

For example, on Linux:

```
export FM_TRACE_PLUGINS=<installation_path>/plugins/<OS_Compiler>/TarmacTrace.so
```

or on Windows:

```
set FM_TRACE_PLUGINS=<installation_path>\plugins\<OS_Compiler>\<version>\TarmacTrace.dll
```

To set multiple trace plug-ins at the same time, separate them with semicolons, for example:

```
set FM_TRACE_PLUGINS=%PVLIB_HOME%\plugins\Win64_VC2019\Release\TarmacTrace.dll;
%PVLIB_HOME%\plugins\Win64_VC2019\Release\GenericTrace.dll
```

You can also load the same plug-in multiple times. Give each instance a name by adding a prefix `<instancename>=` to each plug-in path or paths.

Specify parameters for plug-ins that you load using `FM_TRACE_PLUGINS`, using the following syntax:

```
|<param0_without_prefix=value>||<param1_without_prefix=value>||
<paramN_without_prefix=value>|<absolute_path_to_plugin.dll>
```

where `<param*_without_prefix=value>` means you must specify the parameter without the prefix that would be used on the command line. For example, use `trace-file=file.txt` instead of `TRACE.TarmacTrace.trace-file=file.txt`.

For example, on Linux:

```
export FM_TRACE_PLUGINS='|trace-file=/home/work/trace.txt||end-instruction-
count=1000|$PVLIB_HOME/plugins/Linux64_GCC-7.3/TarmacTrace.so'
```

or on Windows:

```
set FM_TRACE_PLUGINS=^|trace-file=c:\work\trace.txt^|^|end-instruction-count=1000^|
%PVLIB_HOME%\plugins\Win64_VC2019\Release\TarmacTrace.dll
```



Note

- Do not specify a pipe character `|` at the end of the environment variable value.
- Separate individual parameters with two pipe characters `||`, not one.
- Separate the plug-in from the parameters using a single pipe character `|`.
- Do not specify quotes around paths that contain spaces, or escape spaces. Spaces are resolved automatically.

- Do not specify environment variables within this environment variable.
- Specify the parameters before you specify the plug-in.
- On Windows, you can either set the environment variable using the **Advanced System Settings** window, or if you are using the command line, you might need to escape each pipe character, for example `^|`.
- The parameter values that you set using this environment variable are not displayed by the `--list-params` or `-l` command-line options. This is because the environment variable is processed at a later stage, when launching the simulation.

4.2 Customizing a plug-in

You can customize the behavior of a plug-in using parameters.

Plug-in parameters are set in the same way as model parameters. They have the following format:

```
-C <PLUGIN_TYPE>.<plugin_name>.<parameter>=<value>
```

The `PLUGIN_TYPE` value varies depending on the plug-in type, for example:

```
-C TRACE.TarmacTrace.trace-file=trace.log
```

```
-C SVE.ScalableVectorExtension.clear_constrained_lanes=1
```

4.3 ArchMsgTrace

The Architecture Message Trace plug-in prints warning and error messages to `stdout` or to a file when software performs operations that are not recommended, for instance because they are unpredictable.

The plug-in connects to all trace sources that have the `ArchMsg` prefix, which are normal MTI trace sources, but with a specific format. These trace sources can also be used with the `GenericTrace` plug-in, but the `ArchMsgTrace` plug-in has extra capabilities.

When the model emits an `ArchMsg` trace event, `ArchMsgTrace` outputs a message in the format:

```
category: component.hierarchy.name: ...
```

The trace sources have names of the form:

```
component.hierarchy.ArchMsg.category.name[#supplementalEventName]
```

where:

- `category` is usually `Warning`, `Error`, Or `Info`.
- `name` is a short string that uniquely identifies the condition.
- `supplementalEventName` is an optional identifier for a supplemental event, which is an event that provides more information about the initial event. For example, if a cache contains mismatching attributes, which triggers an `ArchMsg` trace event, a supplemental event might be emitted for each cache line affected.

The trace source can also include a line that defines a more human-readable description of the event. This line can contain fields which `ArchMsgTrace` replaces in the output string with values from the trace source.

`ArchMsgTrace` can be configured to suppress:

- Specific trace sources.
- Specific categories.
- Repeated events of the same type.

To suppress specific trace sources or categories, use a whitespace-separated list of patterns, optionally including wildcards (`*` and `?`).

Repeated events can only be suppressed if the `ArchMsg` trace source declares a key field. `ArchMsgTrace` searches for the key field in the following way:

- It looks for the string `"\nPRIMARY KEY <key-field-name>"` in the description of the trace source and uses that field name if the string exists.
- If not found, it looks for a field named `"KEY"`.
- Otherwise, the `ArchMsg` trace source has no key field and cannot be suppressed.

If the `suppress_repeated` parameter is true, the plug-in suppresses repeated events for the same trace source that have the same key field value. For example, the key field might represent the PC and so repeated events for the same PC can be suppressed.



To see a list of all possible `ArchMsg` trace sources that the model can emit, run it with the `ListTraceSources` plug-in. Then search the output for trace sources with the `ArchMsg` prefix.

Some examples of `ArchMsg` trace sources are:

`ArchMsg.Error.BusActiveDuringReset`

A transaction was received at the bus slave port whilst reset was asserted.

`ArchMsg.Warning.cache_contents_unknown`

Execution that depends on unknown cache contents.

`ArchMsg.Warning.warning_atomic_to_unsupported_memory`

Atomic access to an unsupported memory type.

ArchMsg.Warning.decode_unpred_other

Use of unpredictable instruction.

ArchMsg.Warning.recursive_exception

Recursive exception.

Related information[ListTraceSources](#) on page 4174

4.3.1 ArchMsgTrace - parameters

This section describes the parameters for the ArchMsgTrace plug-in.

Each parameter is prefixed with `TRACE.ArchMsgTrace`, for example:

```
TRACE.ArchMsgTrace.exit_on_first_output
```

Table 4-1: ArchMsgTrace parameters

Name	Type	Default value	Allowed values	When set	Description
exit_on_first_output	bool	false	true, false	Init time	Exit the simulation process after the first message has been written.
filter_tags	string	"ALL"	""	Init time	Space-separated list of tags that are matched against the tag(s) of the trace events. The trace event message is printed if any of the tags matches. If the value is empty or ALL, all the messages are printed. Available tags: - ALL - UNPREDICTABLE - IMP_DEF.
suppress_categories	string	"Why"	""	Init time	Space-separated list of categories which should not be printed.
suppress_repeated	bool	true	true, false	Init time	Suppress repeated messages from similar call sites.
suppress_sources	string	""	""	Init time	Space-separated list of components or events that should not be printed.
trace-file	string	""	""	Init time	ArchMsgTrace output file.

4.4 ASTFplugin

ASTFplugin is an MTI plug-in that enables Fast Models to generate trace output in Architectural Structured Trace Format (ASTF).

ASTF is a binary, compressible trace format that captures the architectural execution of each of the CPUs in a system. It supports the collection of traces from complex workloads of up to billions of instructions in length. The format was designed to achieve a balance between compactness, ease of interpretation, and strong forwards and backwards compatibility.

ASTF and associated tools have been developed to support workload tracing, workload analysis, and to drive CPU performance models.

ASTFplugin can be used in combination with [ToggleMTIPlugin](#).



Note

- ASTFplugin is supported on Linux hosts only.
- ASTFplugin and the ASTF specification are in development and further iterations are expected. For the status of ASTFplugin, the version of the specification it supports, and any limitations and known issues, see the Fast Models release notes.

Additional reading

- The ASTF specification is included in the Fast Models package in the `$FVLIB_HOME/Docs/` directory.
- For answers to some common queries about ASTFplugin, see the [ASTFplugin FAQs](#).
- For best practices for preparing an ASTF trace for performance prediction of a workload see [Workload Trace Generation Best Practices](#).
- For how to use a Fast Models FVP to capture an ASTF trace, see [How to generate ASTF traces of workloads running on Fast Models](#).

4.4.1 ASTFplugin - parameters

This section describes the parameters for the ASTFplugin plug-in.

Each parameter is prefixed with `TRACE.ASTFplugin`, for example:

```
TRACE.ASTFplugin.encoding-method
```

Table 4-2: ASTFplugin parameters

Name	Type	Default value	Allowed values	When set	Description
encoding-method	int	0x2	0x0 - 0x2	Runtime	ASTF record encoding method. 0: Uncompressed; 1: Compressed LZMA; 2: Compressed ZLib.
timestamp-enable	bool	false	true, false	Init time	Timestamp records will become part of the output if enabled.
timestamp-period	int	0x5	0x1 - 0x7fffffffffffffffff	Init time	This parameter sets the simulated time between two timestamps in micro-seconds.
trace-file	string	""	""	Runtime	Trace file pathname and prefix to write out to. Will be appended with component path, session number and .astf suffix.
verbosity	int	0x2	0x0 - 0x2	Runtime	Output verbosity level. 0: FATAL; 1: ERROR; 2: WARNING.

4.4.2 ASTFplugin usage notes

Be aware of the following points when using ASTFplugin.

- Load ASTFplugin in the same way as other plug-ins, using the syntax:

```
./isim_system <isim_params> --plugin /path/to/ASTFplugin.so <astf_plugin_params>
```

- ASTFplugin generates trace files with a `.astf` extension. During the simulation, these trace files might be incomplete. Incomplete trace files have a `.astf.part` extension and cannot be processed using the ASTF tools.
- The output trace files have a four digit enumerator field in the name. For example `FVP_Base_Cortex_A55x2.cluster0.cpu0.0000.astf`. This enumerator field is always present, regardless of whether ASTFplugin is used together with ToggleMTIPlugin. If ToggleMTIPlugin instructs ASTFplugin to stop and then resume, a new file is created for each CPU with each enumerator field incremented by one. However, if a CPU was not active when ToggleMTIPlugin instructed ASTFplugin to record, the respective output file is not created.
- If parameter `TRACE.ASTFplugin.timestamp-enable` is set to 1, timing annotation must be enabled, or the plug-in outputs an error message and terminates the simulation. Timing annotation is enabled by default. It is disabled by setting environment variable `FASTSIM_DISABLE_TA` to 1.
- ASTFplugin tries to register callbacks for MTI trace sources for the Scalable Vector Extension (SVE). If `scalableVectorExtension.so` is not enabled, ASTFplugin reports warnings to the console. You can ignore these warnings if SVE operations do not need to be recorded or if SVE is intended to be disabled. To enable SVE operations, load `scalableVectorExtension.so` using the `--plugin` option. To record SVE operations, `scalableVectorExtension.so` must be loaded before `ASTFplugin.so`.
- To improve performance, ASTFplugin is multithreaded. As the plug-in handles large streams of data, avoid using SMT or Hyper-Threading or running the threads on different sockets on a multi-socket host system. For optimal performance, we recommend you use `taskset` to restrict the model to using a specified set of N+1 host cores where N is the number of cores simulated in the model.

4.4.3 Additional ASTF support in Fast Models

In addition to ASTFplugin, Fast Models includes some other tools and libraries that support ASTF.

- The `trprint`, `trcheck`, `trdd`, `trimage`, and `trpidannotate` tools enable you to process the ASTF trace file. For details, see [ASTF tools](#).
- [ToggleMTIPlugin](#), installed in `$PVLIB_HOME/plugins/<OS_Compiler>/`, can be used together with ASTFplugin to limit trace generation to specific regions of interest.
- libastf library and header files:

`$PVLIB_HOME/astf_tools/lib/libastf.a`

Library for reading and writing ASTF trace files. It exposes both C++ and C interfaces. For documentation, including a basic C++ example, see `$PVLIB_HOME/Docs/astf/libastf-api/libastf-api.txt`.

\$PVLIB_HOME/astf_tools/include/astf.h

Specifies the C++ and C interfaces to libastf. For documentation of each API function, see `$PVLIB_HOME/Docs/astf/libastf-api/<C++_function_name>.txt`.

\$PVLIB_HOME/astf_tools/include/astf_records.h

ASTF library record definitions.

- An example Python script, `$PVLIB_HOME/examples/pyIris/inst_count_trace_control.py`. It uses the `iris.debug` Python module to demonstrate using `ToggleMTIPlugin` to limit tracing to specific parts of the application. For usage instructions, run the script with the `-h` option. For more information, see the comments in the source file.

Related information

[Iris Python Debug Scripting User Guide](#)

4.4.4 ASTF tools

The ASTF-related tools `trcheck`, `trdd`, `trimage`, `trpidannotate`, and `trprint` are installed in `$PVLIB_HOME/astf_tools/`. They enable you to process the trace files that `ASTFplugin` outputs, for example to view them in a human-readable format.

trcheck

Verifies the correctness of the trace files against the semantics defined in the ASTF format specification. For documentation, see `$PVLIB_HOME/Docs/astf/tools/trcheck.txt`.

trdd

Slices, copies, and (re)compresses the trace files. It can cut pieces from a trace file or re-encode a trace file by using a different compression level. For documentation, see `$PVLIB_HOME/Docs/astf/tools/trdd.txt`.

trimage

Analyses and profiles instructions and branches across multiple ASTF files. For documentation, see `$PVLIB_HOME/Docs/astf/tools/trimage.txt`.

trpidannotate

Annotates Context records in the trace files to correct the PID/TID information that was collected during tracing. For documentation, see `$PVLIB_HOME/Docs/astf/tools/trpidannotate.txt`.

trprint

Enables viewing and printing trace files in a human-readable format. For documentation, see `$PVLIB_HOME/Docs/astf/tools/trprint.txt`.

4.5 BranchPrediction

The `BranchPrediction` plug-in enables branch prediction modeling in Fast Models. The `BranchPrediction` plug-in is deprecated. It might be modified or removed in a future release.

Branch prediction is an aspect of Timing Annotation. For more details, see [Timing Annotation](#) in the Fast Models User Guide.

The type of branch predictor to use is selected using the `predictor-type` parameter. It can be one of the following example branch predictors that Arm provides, or a user-defined one:

FixedDirectionPredictor

Always takes a preset fixed direction.

BiModalPredictor

Standard 2-bit strength predictor.

GSharePredictor

Standard global history sharing predictor.

HybridPredictor

Selects the majority result from the bimodal predictor, fixed-direction predictor, and a random predictor.

CortexA53Predictor

Cortex®-A53 branch predictor. This is the default.

4.5.1 BranchPrediction - parameters

This section describes the parameters for the `BranchPrediction` plug-in.

Each parameter is prefixed with `BranchPrediction.BranchPrediction`, for example:

```
BranchPrediction.BranchPrediction.bpstat-branchcount
```

Table 4-3: BranchPrediction parameters

Name	Type	Default value	Allowed values	When set	Description
bpstat-branchcount	int	-0x1	0x0 - 0x7fffffffffffffffff	Init time	The number of branch instructions to display. Set to -1 to display all branch instructions.
bpstat-pathfilename	string	""	""	Init time	The path and filename of the branch statistics log.
mispredict-latency	int	0x8	0x0 - 0x7fffffffffffffffff	Init time	The number of instructions that are flushed for every misprediction.
predictor-type	string	"CortexA53Predictor"	""	Init time	The type of branch predictor to use.

4.5.2 BranchPrediction output example

This example command line configures and loads the BranchPrediction plug-in:

```
./EVS_Base_Cortex-A73x1 -a __image.axf \
...
--plugin $PVLIB_HOME/plugins/Linux64_GCC-7.3/BranchPrediction.so \
-C BranchPrediction.BranchPrediction.predictor-type=BiModalPredictor \
-C BranchPrediction.BranchPrediction.mispredict-latency=8 \
-C BranchPrediction.BranchPrediction.bpstat-pathfilename=bpstat.txt \
-C BranchPrediction.BranchPrediction.bpstat-branchcount=5
```

This command produces the following log file:

```
Processor Core: ARM_Cortex-A73
Cluster instance: 0
Core instance: 0
Mispredict Latency: 8
Image executed: __image.axf
PredictorType: BiModalPredictor
Total branch calls: 7757
Total Mispredictions: 130
Average prediction accuracy: 0.983241
Conditional Branches: 139
Total unique branch instructions: 289
--Branch instructions--
  PC Addr      Calls    Mispredict Accuracy
[0] 0x0         2         0           1
[1] 0x80000000  1         0           1
[2] 0x8000000c  10        0           1
[3] 0x80000014  10        0           1
[4] 0x8000001c  10        0           1
```

4.5.3 Other ways to report branch mispredictions

These are some alternative ways to report branch mispredictions.

- The GenericTrace plug-in can generate MTI trace events that report branch mispredictions. For example:

```
--plugin $PVLIB_HOME/plugins/Linux64_GCC-7.3/GenericTrace.so \
-C TRACE.GenericTrace.trace-sources=BRANCH_MISPREDICT
```

The BranchPrediction plug-in must also be loaded to generate these events.

- The BranchPrediction plug-in is integrated with the PMU event counters. On a Linux boot simulation, you can track the number of simulated branch mispredictions in an application by loading the BranchPrediction plug-in and running the `perf` tool. For example, the following command displays the number of branch mispredictions that are made in an application called `testapp`:

```
perf stat e branch-misses testapp
```

4.6 CADIIPCRemoteConnection

Use this plug-in to make a remote CADI connection to a model.

The remote connection can be from a client on a different host machine, or from a client on the same host but using a specific port.

To enable remote connections, set the `enable_remote_cadi` plug-in parameter to true and specify an IP address to listen to, or `0.0.0.0` to listen to all adapters. Also, you must start a CADI server with the `-s` model parameter.

The default values for this plug-in restrict connections to be from the localhost (127.0.0.1) only.

Related information

- [Connect Model Debugger to a model running on another host](#)

4.6.1 CADIIPCRemoteConnection - parameters

This section describes the parameters for the CADIIPCRemoteConnection plug-in.

Each parameter is prefixed with `REMOTE_CONNECTION.CADIIPCRemoteConnection`, for example:

```
REMOTE_CONNECTION.CADIIPCRemoteConnection.enable_remote_cadi
```

Table 4-4: CADIIPCRemoteConnection parameters

Name	Type	Default value	Allowed values	When set	Description
<code>enable_remote_cadi</code>	bool	false	true, false	Init time	Allow connections from remote hosts.
<code>listen_address</code>	string	"127.0.0.1"		Init time	If <code>enable_remote_cadi</code> is set, the network address the server listens on. The default is 127.0.0.1.
<code>port</code>	int	0x7b8b	0x1 - 0xffff	Init time	If <code>enable_remote_cadi</code> is set, the TCP port the server listens on. The default is 31627.
<code>range</code>	int	0x0	0x0 - 0x64	Init time	If the requested port is not available, search for the next available port in the range [port:port+range]. The default is zero, which means only try the requested port.

4.7 CDE

Custom Datapath Extension (CDE) allows you to improve performance and efficiency by adding application domain-specific features to embedded processors, while maintaining the advantages of the Arm® software ecosystem.

CDE allows you to add a customizable module inside some Cortex®-M processors. This module is driven by the pre-decoded CDE instructions and shares the same interface as the standard Arithmetic Logic Unit (ALU) of the processor.

Fast Models implements CDE using Model Trace Interface (MTI) plug-ins, with CADI parameters to allow the plug-ins to be configured at runtime. The following Fast Models support CDE:

- [ARMCortexM33CT](#)
- [ARMCortexM52CT](#)
- [ARMCortexM55CT](#)
- [ARMCortexM85CT](#)
- [ARMAEMv8MCT](#)

The following model parameters are exposed for configuring CDE:

has_cde

Controls whether CDE is enabled. If enabled, a plug-in must be provided.

--plugin <path/to/plugin.so>

This option can be specified multiple times, once for each CDE plug-in implementation. Alternatively, plug-ins can be loaded by setting the `FM_PLUGINS` or `FM_TRACE_PLUGINS` environment variable.

cpu.cde_impl_name=<plugin_name>

The CDE implementation name to use with this core. If multiple CDE plug-in implementations are provided, each core can be requested to use a specific plug-in by using `cpu<n>.cde_impl_name=...`

Two example plug-ins are available, CDETester and CDEConstant. They are provided as pre-built libraries and as source code, located in `$PVLIB_HOME/plugins/source/`, to help with implementing your own plug-ins.

4.7.1 CDETester

CDETester is a basic example plug-in that allows you to specify at runtime which CDE instructions are supported by individual coprocessors and to specify the behavior, either nop or undefined.

To specify the instructions that coprocessors support, provide a plug-in parameter of the form:

```
CDE.CDETester.cde_tester_trivial.cps_implemented_instr=0xn
```

where `0xn` represents a hexadecimal bitmask of coprocessors that implement this instruction and `instr` represents a CDE instruction name.

The full list of CDE instruction names is as follows, where `a` represents dual variants and `v` represents vector variants:

- `cx1`
- `cx2`
- `cx3`
- `cx1d`
- `cx2d`
- `cx3d`
- `vcx1`
- `vcx2`
- `vcx3`
- `vcx1v`
- `vcx2v`
- `vcx3v`

Accumulate variants are handled in the same function implementation as the non-accumulate variants.

The bitmask takes the form:

bits [7:0]

For non-accumulate variants.

bits [23:16]

For accumulate variants.

The plug-in also allows control over which coprocessors implement CDE through the `CDE.CDETester.cde_tester_trivial.cps_implemented=0xn` parameter, where `0xn` represents a hexadecimal bitmask of coprocessors that implement CDE.

An example invocation to enable `cx1` and `cx1A` (accumulate) for CP3 might be written as:

```
CDE.CDETester.cde_tester_trivial.cps_implemented=0x8 // enable CDE support for CP3
CDE.CDETester.cde_tester_trivial.cps_implemented_cx1=0x80008 // enable CX1 and CX1A
on CP3 (behaves as nop rather than undef)
```

On Linux, you can retrieve an up-to-date list of parameters from the model by using:

```
-l | grep -i cdetester
```

4.7.2 CDETester - parameters

This section describes the parameters for the CDETester plug-in.

Each parameter is prefixed with `cde.cdeTester`, for example:

```
cde.cdeTester.cde_tester_trivial.cps_implemented
```

Table 4-5: CDETester parameters

Name	Type	Default value	Allowed values	When set	Description
<code>cde_tester_trivial.cps_implemented</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Bitmask indicating coprocessors implemented.
<code>cde_tester_trivial.cps_implemented_cx1</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables CX1 instructions ([23:16] for CX1A, [7:0] for CX1).
<code>cde_tester_trivial.cps_implemented_cx1d</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables CX1D instructions ([23:16] for CX1DA, [7:0] for CX1D).
<code>cde_tester_trivial.cps_implemented_cx2</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables CX2 instructions ([23:16] for CX2A, [7:0] for CX2).
<code>cde_tester_trivial.cps_implemented_cx2d</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables CX2D instructions ([23:16] for CX2DA, [7:0] for CX2D).
<code>cde_tester_trivial.cps_implemented_cx3</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables CX3 instructions ([23:16] for CX3A, [7:0] for CX3).
<code>cde_tester_trivial.cps_implemented_cx3d</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables CX3D instructions ([23:16] for CX3DA, [7:0] for CX3D).
<code>cde_tester_trivial.cps_implemented_vcx1</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables VCX1 instructions ([23:16] for VCX1A, [7:0] for VCX1).
<code>cde_tester_trivial.cps_implemented_vcx1v</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables VCX1 (Vector) instructions ([23:16] for VCX1A (Vector), [7:0] for VCX1 (Vector)).
<code>cde_tester_trivial.cps_implemented_vcx2</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables VCX2 instructions ([23:16] for VCX2A, [7:0] for VCX2).
<code>cde_tester_trivial.cps_implemented_vcx2v</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables VCX2 (Vector) instructions ([23:16] for VCX2A (Vector), [7:0] for VCX2 (Vector)).
<code>cde_tester_trivial.cps_implemented_vcx3</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables VCX3 instructions ([23:16] for VCX3A, [7:0] for VCX3).
<code>cde_tester_trivial.cps_implemented_vcx3v</code>	int	0xff00ff	0x0 - 0xffffffff	Init time	Coprocessor enables VCX3 (Vector) instructions ([23:16] for VCX3A (Vector), [7:0] for VCX3 (Vector)).
<code>has_cde_tester_trivial</code>	bool	true	true, false	Init time	Whether the CDETester plugin is implemented (undefs all CDE instructions).

4.7.3 CDEConstant

This is an example plug-in that provides an implementation for every CDE instruction variant, for example [v]cxn, [v]cxnA, and [v]cxnD. Each instruction simply performs an XOR with some arguments and a constant.

To load this plug-in, use the following parameters when launching the model:

```
--plugin path/to/plugin/CDEConstant.so -C cpu.has_cde=1 -C
cpu.cde_impl_name=CDE_CONSTANT
```

4.7.4 CDEConstant - parameters

This section describes the parameters for the CDEConstant plug-in.

Each parameter is prefixed with `CDE.CDEConstant`, for example:

```
CDE.CDEConstant.has_cde_constant
```

Table 4-6: CDEConstant parameters

Name	Type	Default value	Allowed values	When set	Description
has_cde_constant	bool	true	true, false	Init time	Whether the CDEConstant plugin is implemented.

4.7.5 Implementing a CDE plug-in

Fast Models supports prototyping of custom instructions through a modular plug-in system, which uses the Model Trace Interface (MTI) framework.

Multiple CDE plug-ins can be registered with the model and each core can be instructed which plug-in behavior to use. Run-time configuration of CDE plug-ins is performed using CADI parameters, although plug-in developers can use alternative approaches, for example configuration files.

This guide shows how to implement a basic MTI-based CDE plug-in, using the CDETester plug-in as an example. It is intended to be a quick start to plug-in development, and does not describe details about MTI. To learn more about MTI, see [Model Trace Interface Reference Manual](#). The source code for CDETester can be found in `$PVLIB_HOME/plugins/source/CDETester/`.

A CDE plug-in performs three main tasks:

- MTI and CDE interface registration.
- Handling of parameters passed through the command line or CADI.
- Implementing CDE instructions.

Related information

- [CDETester](#)

4.7.5.1 Prerequisites for implementing a CDE plug-in

To build the CDE plug-in examples, you need the following:

- A compiler that matches the Fast Models build you are using.
- A recent version of CMake.
- An installation of the Fast Models package and libraries.

4.7.5.2 CDE plug-in registration

After a CDE plug-in has been loaded into the model through the `--plugin` argument, or the `FM_TRACE_PLUGINS` environment variable, it must register itself with the CDE interface registry using the MTI framework API.

The CDE interface registry is responsible for:

- Managing all loaded CDE plug-ins.
- Passing any parsed arguments to the relevant CDE plug-in.
- Assigning CDE plug-ins to cores, as requested by the model arguments.

The CDE interface registry requires an interface name and version to be registered through MTI, as shown in `CDETester.cpp` and `CDETesterTrivialImpl.h`.

4.7.5.3 CDE plug-in parameters

After the plug-in has registered itself with the CDE interface registry, the model passes any parsed command-line parameters to the CDE plug-in they are associated with.

The parameters to the `CDETester` example plug-in allow you to specify which custom instructions result in a no operation (nop) for each coprocessor. Any instructions that are not enabled result in an Undefined Instruction exception being raised.

The plug-in handles parameters that it receives from the model in `CDETester.cpp` by passing them through the `CDETesterFactory` interface to the handler implementation in `CDETrivialImpl::consumeParameter()`, defined in `CDETesterTrivialImpl.cpp`.

Related information

- [CDETester](#)

4.7.5.4 CDE plug-in instruction handler interface

After plug-in registration and optional parameter handling, the plug-in should implement handlers for all available CDE instructions, even if it does not intend to implement custom functionality for all instruction variants.

To help you do this, Fast Models provides a utility header, `$PVLIB_HOME/include/ct/CDE/CDEInstHandlerInterface.h`. This header defines the `CDEInstHandlerInterface` class for handling CDE instruction calls, which plug-ins must inherit.

This interface declares pure virtual methods for each CDE instruction variant, for example dual and accumulator, as well as the structures containing decode information and call results. A typical signature follows this pattern:

```
CDEResult64 CDETrivialImpl::exec_cx2_d(const CX2DecodeInfo& decode_info, uint64_t
rfd_val, uint32_t rn_val)
```

The `CDETester` example plug-in inherits this interface in `CDETesterBaseImpl.h` and implements the instructions in `CDETesterTrivialImpl.h` and `CDETesterTrivialImpl.cpp`.

4.7.5.5 CDE plug-in instruction implementations

Each CDE instruction is mapped to a single function definition, except for accumulate variants, which are handled by checking for the accumulate flag in the parameters passed to the instruction implementations.

The full list of instructions that a plug-in is expected to implement is given in [CDETester](#).

Each instruction implementation accepts as parameters:

- A structure containing decoded instruction opcode parameters, including register numbers and immediate. For example `CX1DecodeInfo`.
- The contents of registers specified in the instruction opcode.

Instruction implementations should return a result structure of varying size, for example `CDEResult32` or `CDEResult64`. This structure indicates whether the instruction is supported, and if so, the return value and the number of cycles taken for execution, which is used in trace and performance analysis.

If a plug-in needs to raise an Undefined Instruction exception for a particular instruction, it can simply return a default-initialized result structure.

For full details of the expected function declarations and parameter types, see the file `$PVLIB_HOME/include/ct/CDE/CDEInstHandlerInterface.h`.

4.7.5.6 Building the CDE plug-in

During plug-in development, either modify the example `CMakeLists.txt` files provided with the CDE plug-ins to reflect any changes to the file structure, or alternatively, use your own build system.

To build the example plug-ins:

1. Run `cmake` on the root directory of the plug-in to generate the project file, for example a Makefile or Visual Studio solution.
2. Run `make`.

4.7.6 CDE API

Reference documentation for the CDE API.

The CDE API is defined in the following header files, which are located in `$PVLIB_HOME/include/ct/CDE/`:

- `CDEFactoryInterface.h`
- `CDEInstHandlerInterface.h`
- `CDERegistryInterface.h`

4.7.6.1 CDEFactoryInterface.h

Defines the interface for obtaining an instance of a `CDEInstHandlerInterface` for a specific core.

4.7.6.1.1 CDE::CDEFactoryInterface class

The factory interface for obtaining an instance of a `CDEInstHandlerInterface` for a specific core.

4.7.6.1.2 CDE::CDEFactoryInterface::instantiateCDEInstHandler()

Instantiate a `CDEInstHandlerInterface` for a given core. The caller owns the result.

```
virtual std::unique_ptr<CDEInstHandlerInterface>
    instantiateCDEInstHandler(std::string component_hierarchy) = 0;
```

component_hierarchy

String representing the hierarchy of the current component.

4.7.6.1.3 CDE::CDEFactoryInterface::CDEImplName()

Return the name of the CDE implementation. A core can use this method to disambiguate multiple CDE implementations in a simulation.

```
virtual std::string CDEImplName() const = 0;
```

4.7.6.1.4 CDE::CDEFactoryInterface::CDEImplDescription()

Description of the CDE implementation.

Return the description of the CDE implementation that can be instantiated.

```
virtual std::string CDEImplDescription() const = 0;
```

4.7.6.1.5 CDE::CDEFactoryInterface::CDEImplProviderName()

Provider name of the CDE implementation.

Return the name of the component providing the CDE implementation, for example a plug-in. This name might be used in informative diagnostic messages.

```
virtual std::string CDEImplProviderName() const = 0;
```

4.7.6.2 CDEInstHandlerInterface.h

Defines the interface for executing CDE instructions.

4.7.6.2.1 CDE::CDEResult32 struct

32-bit result of a CDE instruction.

Members

instr_not_supported

Whether the instruction is supported.

value

Return value of the instruction.

cycles

Number of cycles of instruction execution.

4.7.6.2.2 CDE::CDEResult64 struct

64-bit result of a CDE instruction.

Members

instr_not_supported

Whether the instruction is supported.

value

Return value of the instruction.

cycles

Number of cycles of instruction execution.

4.7.6.2.3 CDE::CDEResult128 struct

128-bit result of a CDE instruction.

Members

instr_not_supported

Whether the instruction is supported.

value_lo

Low 64 bits of the return value of the instruction.

value_hi

High 64 bits of the return value of the instruction.

cycles

Number of cycles of instruction execution.

4.7.6.2.4 CDE::CX1DecodeInfo struct

Decoded information for the cx1 instruction.

Members

accumulate

Whether to accumulate with existing register contents.

coproc

Number of coproc.

imm

Immediate value.

rd_num

General-purpose destination register number.

4.7.6.2.5 CDE::CX2DecodeInfo struct

Decoded information for the cx2 instruction.

Members

accumulate

Whether to accumulate with existing register contents.

coproc

Number of coproc.

imm

Immediate value.

rd_num

General-purpose destination register number.

rn_num

General-purpose source register number.

4.7.6.2.6 CDE::CX3DecodeInfo struct

Decoded information for the cx3 instruction.

Members

accumulate

Whether to accumulate with existing register contents.

coproc

Number of coproc.

imm

Immediate value.

rd_num

General-purpose destination register number.

rn_num

General-purpose source register number.

rm_num

General-purpose source register number.

4.7.6.2.7 CDE::VCX1DecodeInfo struct

Decoded information for the vcx1 instruction.

Members

accumulate

Whether to accumulate with existing register contents.

coproc

Number of coproc.

imm

Immediate value.

vd_num

Source and destination vector register number.

4.7.6.2.8 CDE::VCX2DecodeInfo struct

Decoded information for the vcx2 instruction.

Members

accumulate

Whether to accumulate with existing register contents.

coproc

Number of coproc.

imm

Immediate value.

vd_num

Source and destination vector register number.

vm_num

Source vector register number.

4.7.6.2.9 CDE::VCX3DecodeInfo struct

Decoded information for the vcx3 instruction.

Members

accumulate

Whether to accumulate with existing register contents.

coproc

Number of coproc.

imm

Immediate value.

vd_num

Source and destination vector register number.

vn_num

Source vector register number.

vm_num

Source vector register number.

4.7.6.2.10 CDE::CDEInstHandlerInterface class

Interface for executing CDE instructions.

This class defines the following methods for executing CDE instructions:

exec_cx1()

cx1 instruction.

exec_cx1_d()

cx1D instruction.

exec_cx2()

cx1 instruction.

exec_cx2_d()

cx2D instruction.

exec_cx3()

cx3 instruction.

exec_cx3_d()

cx3D instruction.

exec_vcx_1_s()

vcx1 instruction with S register.

exec_vcx_1_d()

vcx1 instruction with D register.

exec_vcx_1_q()

vcx1 instruction with Q register.

exec_vcx_2_s()

vcx2 instruction with S register.

exec_vcx_2_d()

vcx2 instruction with D register.

exec_vcx_2_q()

vcx2 instruction with Q register.

exec_vcx_3_s()

vcx3 instruction with S register.

exec_vcx_3_d()

vcx3 instruction with D register.

exec_vcx_3_q()

vcx3 instruction with Q register.

exec_vcx_1_beatwise()

vcx1 instruction for one beat. Caller handles predicated writeback.

exec_vcx_2_beatwise()

vcx2 instruction for one beat. Caller handles predicated writeback.

exec_vcx_3_beatwise()

vcx3 instruction for one beat. Caller handles predicated writeback.

4.7.6.2.11 CDE::CDEInstHandlerInterface::getCDECoprocessorMask()

Return a bitmask indicating which coprocessor numbers this CDE implementation subsumes.

```
virtual uint8_t getCDECoprocessorMask() = 0;
```

4.7.6.3 CDERegistryInterface.h

Defines the interface to allow components, for instance plug-ins, to contribute CDE implementations to the simulation.

4.7.6.3.1 CDE::CDERegistryInterface class

Interface to register the CDE factory.

```
class CDERegistryInterface : public eslapi::CAInterface
```

This class is the interface to register the CDE factory into the Fast Models simulation component registry.

4.7.6.3.2 CDE::CDERegistryInterface::registerCDEFactory()

Register the CDE factory with the simulation.

```
virtual bool registerCDEFactory(std::ostream& error_stream, CDEFactoryInterface*  
interface) = 0;
```

error_stream

The error stream.

interface

The CDE factory interface used to register.

4.7.6.3.3 CDE::CDERegistryInterface::unregisterCDEFactory()

Unregister the CDE factory from the simulation.

```
virtual void unregisterCDEFactory(CDEFactoryInterface* interface) = 0;
```

interface

The CDE factory interface used to unregister.

4.7.6.3.4 CDE::CDERegistryInterface::sendToCores()

Instantiate a CDEInstHandler and send it to the core.

The core can then call the CDE instruction execution functions provided by the plug-in on that CDEInstHandler.

```
virtual bool sendToCores() = 0;
```

4.8 Crypto

The `crypto` plug-in enables Armv8 and Armv9 processor models to support the Armv8.0 Cryptographic Extensions and Armv8.3 architected Pointer Authentication algorithms.

The `crypto` plug-in is available for download from the [Arm Developer website](#).

When the plug-in is loaded:

- All Armv8 and Armv9 processors in the system implement all functionality from the Armv8.0 Cryptographic Extensions by default, although you can disable it for specific processors by setting their `CRYPTODISABLE` parameter.
- All Armv8.3 and Armv9 processors in the system implement the architected algorithms for Pointer Authentication and Generic Authentication by default. Plug-in parameters control which processors in the system have architected algorithms enabled.

AEMs, for example, AEMvACT, have parameters that allow you to restrict the `crypto` plug-in features. These parameters use the same encodings as the flags within the AArch32 `ID_ISAR5` and AArch64 `ID_AA64ISAR0_EL1` system registers. You can set these parameters for a specific AEM core using this syntax:

`-C cpu.cpu<X>.<feature_name>=<value>`

Where `feature_name` can be one of the following:

- `crypto_aes`, with these possible values:
 - 0**
No AES instructions are implemented
 - 1**
The `AESE`, `AESD`, `AESMC`, and `AESIMC` instructions are implemented
 - 2**
As 1, but in addition, the `PMULL` and `PMULL2` instructions can operate on 64-bit data values. This is the default value.
- `crypto_sha1`, with these possible values:
 - 0**
No SHA-1 instructions are implemented
 - 1**
The `SHA1C`, `SHA1P`, `SHA1M`, `SHA1H`, `SHA1SU0`, and `SHA1SU1` instructions are implemented. This is the default value.
- `crypto_sha256`, with these possible values:
 - 0**
No SHA-256 instructions are implemented
 - 1**
The `SHA256H`, `SHA256H2`, `SHA256SU0`, and `SHA256SU1` instructions are implemented. This is the default value.
- `crypto_sha3`, with these possible values:
 - 0**
No Armv8.4 SHA-3 instructions are implemented. This is the default value.
 - 1**
SHA-3 instructions are implemented if Armv8.4 is enabled.
 - 2**
SHA-3 instructions are implemented.
- `crypto_sha512`, with these possible values:
 - 0**
No Armv8.4 SHA-512 instructions are implemented. This is the default value.
 - 1**
SHA-512 instructions are implemented if Armv8.4 is enabled.
 - 2**
SHA-512 instructions are implemented.
- `crypto_sm3`, with these possible values:

0

No Armv8.4 SM-3 instructions are implemented. This is the default value.

1

SM-3 instructions are implemented if Armv8.4 is enabled.

2

SM-3 instructions are implemented.

- `crypto_sm4`, with these possible values:

0

No Armv8.4 SM-4 instructions are implemented. This is the default value.

1

SM-4 instructions are implemented if Armv8.4 is enabled.

2

SM-4 instructions are implemented.

For example, to disable the AES instructions on core 0:

```
./isim_system --plugin Crypto.so -C cpu.cpu0.crypto_aes=0
```



These parameters are only available for AEMs. For other Armv8-A and Armv9-A models, the behavior is fixed to the default values.

4.8.1 Crypto - parameters

This section describes the parameters for the Crypto plug-in.

Each parameter is prefixed with `CRYPTO.Crypto`, for example:

```
CRYPTO.Crypto.authentication_algorithm
```

Table 4-7: Crypto parameters

Name	Type	Default value	Allowed values	When set	Description
authentication_algorithm	string	"QARMA5"	""	Init time	Choice of PACAlgorithm. Valid values: "QARMA5", "QARMA3". "QARMA3" can be enabled only if the core feature <code>has_qarma3_pac</code> is true. Default value: "QARMA5". The pattern is a comma separated list of glob-style patterns. These patterns are matched against the instance name of cores in the system. If a pattern matches then the Architected algorithm is installed on the core if that core supports ARMv8.3. For example: - "*", install the algorithm on all supported cores - "cluster0.*", install the algorithm on all cores in cluster0 - "cluster0.cpu0" install the algorithm only on cluster0.cpu0 - "cluster0.*,cluster1.cpu1" install the algorithm on all cores in cluster0 and also cluster1.cpu1 - "" (an empty string), do not install the algorithm on any core .

Name	Type	Default value	Allowed values	When set	Description
generic_authentication_core_pattern	string	"*"	""	Init time	install the ARMv8.3 Architected Generic Authentication algorithm only on ARMv8.3 cores matching one of these patterns. The pattern is a comma separated list of glob-style patterns. These patterns are matched against the instance name of cores in the system. If a pattern matches then the Architected algorithm is installed on the core if that core supports ARMv8.3. For example: - "*", install the algorithm on all supported cores - "cluster0.*", install the algorithm on all cores in cluster0 - "cluster0.cpu0" install the algorithm only on cluster0.cpu0 - "cluster0.*,cluster1.cpu1" install the algorithm on all cores in cluster0 and also cluster1.cpu1 - "" (an empty string), do not install the algorithm on any core .
pointer_authentication_core_pattern	string	"*"	""	Init time	install the ARMv8.3 Architected Pointer Authentication algorithm only on ARMv8.3 cores matching one of these patterns. The pattern is a comma separated list of glob-style patterns. These patterns are matched against the instance name of cores in the system. If a pattern matches then the Architected algorithm is installed on the core if that core supports ARMv8.3. For example: - "*", install the algorithm on all supported cores - "cluster0.*", install the algorithm on all cores in cluster0 - "cluster0.cpu0" install the algorithm only on cluster0.cpu0 - "cluster0.*,cluster1.cpu1" install the algorithm on all cores in cluster0 and also cluster1.cpu1 - "" (an empty string), do not install the algorithm on any core .
verbose	int	0x0	0x0 - 0x1	Init time	verbosity level. 0, terse. 1, verbose.

4.9 GDBRemoteConnection

The `GDBRemoteConnection` plug-in allows the model to be debugged using GDB. It is included in the Third Party IP add-on package to Fast Models.

For more details about the add-on package, contact [Arm Technical Support](#).

Load the plug-in using the following command-line options:

- `--allow-debug-plugin`, or the short version, `-D`
- `--plugin`

For example:

```
./isim_system --allow-debug-plugin --plugin GDBRemoteConnection.so
```

Then, a suitable GDB can be connected to the model using the GDB `target` command.

`GDBRemoteConnection` supports the following operations:

- Connection to models that contain a single core cluster.
- Read and write of core registers.
- Read and write of memory.
- Run, stop, single step.

- Breakpoints.
- Connection to AArch32 and AArch64 models.

4.9.1 GDBRemoteConnection limitations

This section describes the limitations of the `GDBRemoteConnection` plug-in.

- Connections are only allowed to single-core simulations, not to multiprocessor or multicluster simulations.
- No tracepoint support.
- No parameter support.
- The memory view only shows the current memory space.
- The disassembly only uses the current instruction set.
- Breakpoints can only be set on the current memory space.

You cannot use GDB when debugging software that uses semihosting. When a program tries to use semihosting with the GDB plug-in, the GDB debugger wrongly reports having hit a breakpoint:

```
Program received signal SIGTRAP, Trace/breakpoint trap.
```

When using the `GDBRemoteConnection` plug-in, start the simulation with the `-D` (or `-s`) flag. Attaching the `GDBRemoteConnection` plug-in to a running simulation can cause segmentation faults on the simulation and the GDB client.

4.10 GenericCounter

`GenericCounter` is an example MTI plug-in that prints to stdout the number of occurrences of a specific trace source when the simulation terminates.

`GenericCounter` only counts a single trace source, which is set using the `TRACE.GenericCounter.trace-source` parameter. To count multiple trace sources, load the plug-in multiple times. In this case, each plug-in instance has a unique name which you use instead of `GenericCounter` when setting its parameters. The names are either set implicitly or explicitly. This example sets them explicitly as `counter1` and `counter2`:

```
--plugin counter1=path/to/GenericCounter.so \  
--plugin counter2=path/to/GenericCounter.so \  
-C TRACE.counter1.trace-source=EXCEPTION \  
-C TRACE.counter2.trace-source=READ_ACCESS
```

Otherwise, each plug-in instance has an implicit name which consists of the plug-in name and a sequential suffix, for example `GenericCounter`, `GenericCounter0`, `GenericCounter1` and so on. This example uses the implicit names:

```
--plugin path/to/GenericCounter.so --plugin path/to/GenericCounter.so \
-C TRACE.GenericCounter.trace-source=EXCEPTION \
-C TRACE.GenericCounter0.trace-source=READ_ACCESS
```

The source code for this plug-in is provided in `$PVLIB_HOME/examples/MTI/GenericCounter/`.

4.10.1 GenericCounter - parameters

This section describes the parameters for the `GenericCounter` plug-in.

Each parameter is prefixed with `TRACE.GenericCounter`, for example:

```
TRACE.GenericCounter.print_on_event
```

Table 4-8: GenericCounter parameters

Name	Type	Default value	Allowed values	When set	Description
<code>print_on_event</code>	string	""	""	Init time	If set, print the count information to stdout when <code>print_on_event</code> trace source fires. If empty, only print the count information at the end of the simulation or when the <code>print_stats</code> parameter is written to.
<code>print_stats</code>	int	0x0	0x0 - 0x0	Runtime	On write, print count information to stdout.
<code>trace-source</code>	string	"INST"	""	Init time	The trace source to be counted. Example: <code>BRA_DIR</code> .

4.11 GenericTrace

`GenericTrace` is a flexible MTI plug-in that allows you to configure which events are traced, using a comma-separated list of trace sources. Output can be printed to a file or to the console.

Specify one or more trace sources using the `trace-sources` parameter, for example:

```
./FVP_Base_AEMvA \
--plugin $PVLIB_HOME/plugins/Linux64_GCC-7.3/GenericTrace.so \
-C TRACE.GenericTrace.trace-sources=EXCEPTION,EXCEPTION_RETURN
```



To see a list of the available trace sources for each component in the model that provides trace, run the model with the `ListTraceSources` plug-in. See [ListTraceSources](#) for details.

The `trace-sources` parameter provides flexibility:

- To specify trace sources that match a pattern, use the * or ? wildcards, for example:

```
TRACE.GenericTrace.trace-sources=CACHE_*
```

- To trace a specific component only, specify the hierarchical path to it, optionally with wildcards, for example:

```
TRACE.GenericTrace.trace-sources=FVP_Base_AEMvA.cluster0.cpu1.*
```

- To trace specific fields in a trace source, append a field mask. For example to trace only the second field (pc) of the INST trace source, use:

```
TRACE.GenericTrace.trace-sources=INST=0x2
```

Or, to filter out the 8th field (ELEMENT_SIZE) from the CORE_STORES trace source, use:

```
TRACE.GenericTrace.trace-sources=CORE_STORES=0xFFFF7F
```

- If no trace sources are specified, GenericTrace by default traces all the instructions.

The source code for this plug-in is provided as a programming example in `$PVLIB_HOME/examples/MTI/GenericTrace/source/`.



This plug-in can be used with [ToggleMTIPlugin](#).

4.11.1 GenericTrace - parameters

This section describes the parameters for the GenericTrace plug-in.

Each parameter is prefixed with `TRACE.GenericTrace`, for example:

```
TRACE.GenericTrace.collect-latest-data-only
```

Table 4-9: GenericTrace parameters

Name	Type	Default value	Allowed values	When set	Description
collect-latest-data-only	bool	false	true, false	Init time	Collect only latest N trace source fires and print upon destruction.
enabled	bool	true	true, false	Runtime	If set to true, tracing is enabled.
flush	bool	false	true, false	Runtime	If set to true, the trace file is flushed after every event. This has a performance impact but could be used to better debug crashes.

Name	Type	Default value	Allowed values	When set	Description
hide-fieldnames	bool	false	true, false	Runtime	Do not print field names when printing trace output.
latest-data-size	int	0x1	0x0 - 0x7fffffffffffffffff	Init time	Size of latest data to store in buffer for capturing only the latest trace source fires.
perf-period	int	0x0	0x0 - 0x7fffffffffffffffff	Init time	Print performance information every N instructions. 0 means disabled.
print-timestamp	bool	false	true, false	Runtime	Start each trace entry with the host time.
shorten-paths	bool	true	true, false	Runtime	If set to true, the component paths of trace events are shortened by removing the common prefix. The minimal, non-ambiguous path suffix remains. If all traced sources belong to the same components, no path is logged. Default is true.
simulated-timestamp	bool	false	true, false	Runtime	Start each trace entry with the simulated time.
start-icount	int	0x0	0x0 - 0x7fffffffffffffffff	Init time	Start tracing on a certain instruction count. Tracing starts up to 2048 instructions before this count.
stop-icount	int	0x7fffffffffffffffff	0x0 - 0x7fffffffffffffffff	Init time	Stop tracing on a certain instruction count. Tracing stops up to 2048 instructions after this count.
stop_on_event	bool	false	true, false	Init time	Stop the simulation when any event is triggered.
trace-file	string	""	""	Runtime	The trace file to write into. If STDERR, prints to stderr. If empty, prints to stdout.
trace-file-limit	int	0x0	0x0 - 0x7fffffffffffffffff	Init time	The limit of the size of the output file in bytes. The simulation is stopped when this size is reached. If 0, it is unlimited.
trace-sources	string	"INST"	""	Runtime	A comma-separated list of trace sources to be traced. A component path can be prepended, with components separated by dots. Both the component path and the trace source name can contain the wildcards * and ?. A field mask as a number in hex or decimal format can be appended with =. Example: my.subsystem.core.cpu*.TRACE_SOURCE=0x08.
verbose	bool	false	true, false	Runtime	Print some debugging information.

4.11.2 GenericTrace plug-in usage example

This example shows how to use the GenericTrace plug-in to trace accesses by the graphics driver to the registers of a GPU register model.

Procedure

1. Use the ListTraceSources plug-in to list the trace sources that the platform provides. It is located in `$PVLIB_HOME/plugins/<OS_compiler>/:`

```
${PATH_Model} --plugin $PVLIB_HOME/plugins/Linux64_GCC-9.3/ListTraceSources.so
```

The plug-in prints the following information to the terminal:

- All components that provide trace, including the GPU model, for example:

```
Component (292) providing trace: Kits3_Subsys.css.gpu
```

- The trace sources provided by the GPU model. For example, these are some generic trace sources provided by Mali™ GPU models:

INFO_ReadRegister

Access time, addresses, data, and names of the registers that were read.

INFO_Reset

GPU reset data.

INFO_WriteRegister

Access time, addresses, and names of the registers that were updated, and the data before and after the update.

INFO_IrqGpuControl

ID, name, and state of the IRQ signal from the GPU. The state can be `y` for `set`, or `n` for `clear`.

INFO_IrqJobControl

ID, name, and state of the IRQ signal from the Job Manager on the GPU. The state can be `y` for `set`, or `n` for `clear`.

INFO_IrqMmuControl

ID, name, and state of the IRQ signal from the MMU on the GPU. The state can be `y` for `set`, or `n` for `clear`.

WARN_ReadToWriteOnlyRegister

Warning messages and addresses for the write-only registers that have been read by the graphics driver.

WARN_WriteToReadOnlyRegister

Warning messages and addresses for the read-only registers that have been written by the graphics driver.

WARN_AccessToUnimplementedRegister

Warning messages and addresses for the invalid registers that have been accessed by the graphics driver.

- To trace all events from the GPU model, launch the platform with the following additional options:

```
--plugin $PVLIB_HOME/plugins/Linux64_GCC-9.3/GenericTrace.so \
-C TRACE.GenericTrace.trace-sources=Kits3_Subsys.css.gpu.* \
-C TRACE.GenericTrace.enabled=1 \
-C TRACE.GenericTrace.verbose=1 \
-C TRACE.GenericTrace.print-timestamp=1 \
-C TRACE.GenericTrace.trace-file=dp-trace-generic.log
```

Where:

- Kits3_Subsys.css.gpu is the GPU model listed in Step 1.
- Kits3_Subsys.css.gpu.* means trace all the GPU-supported trace sources. Alternatively, (not shown in this example):
 - To trace one specific GPU trace source only, add it as a suffix to the GPU model. For instance, Kits3_Subsys.css.gpu.INFO_ReadRegister.
 - To trace multiple specific trace sources, use a comma-separated list. For instance, Kits3_Subsys.css.gpu.INFO_ReadRegister, Kits3_Subsys.css.gpu.INFO_WriteRegister.
- The trace-file option specifies the log file in which to save the trace output. If it is not used, the trace results are shown on the host terminal.

Results

The host terminal or the log file shows details about the driver-accessed registers, such as the register addresses, data, and the access time. For example:

```
HOST TIME=1557460.545195s INFO_ReadRegister: REG_OFFSET=0x0000000000000000
VALUE=0x60000000 REG_NAME="GPU_ID"
HOST TIME=1557460.545266s INFO_ReadRegister: REG_OFFSET=0x0000000000000004
VALUE=0x07130206 REG_NAME="L2_FEATURES"
HOST TIME=1557460.545279s INFO_ReadRegister: REG_OFFSET=0x0000000000000008
VALUE=0x00000000 REG_NAME="SUSPEND_SIZE"
HOST TIME=1557460.545291s INFO_ReadRegister: REG_OFFSET=0x000000000000000c
VALUE=0x000000809 REG_NAME="TILER_FEATURES"
HOST TIME=1557460.545303s INFO_ReadRegister: REG_OFFSET=0x0000000000000010
VALUE=0x00000001 REG_NAME="MEM_FEATURES"
HOST TIME=1557460.545316s INFO_ReadRegister: REG_OFFSET=0x0000000000000014
VALUE=0x000002830 REG_NAME="MMU_FEATURES"
HOST TIME=1557460.545325s INFO_ReadRegister: REG_OFFSET=0x0000000000000018
VALUE=0x000000ff REG_NAME="AS_PRESENT"
HOST TIME=1557460.545334s INFO_ReadRegister: REG_OFFSET=0x000000000000001c
VALUE=0x00000007 REG_NAME="JS_PRESENT"
HOST TIME=1557460.545345s INFO_ReadRegister: REG_OFFSET=0x00000000000000c0
VALUE=0x00000020e REG_NAME="JS0_FEATURES"
HOST TIME=1557460.545362s INFO_ReadRegister: REG_OFFSET=0x00000000000000c4
VALUE=0x0000001fe REG_NAME="JS1_FEATURES"
HOST TIME=1557460.545364s INFO_ReadRegister: REG_OFFSET=0x00000000000000c8
VALUE=0x00000007e REG_NAME="JS2_FEATURES"
HOST TIME=1515565849.690948s gpu.INFO_WriteRegister: REG_OFFSET=0x0000000000001870
VALUE=0x00000000 UPDATED_VALUE=0x00000000 REG_NAME="JOB_SLOT0_JS_FLUSH_ID_NEXT"
HOST TIME=1515565849.691304s gpu.INFO_WriteRegister: REG_OFFSET=0x0000000000001860
VALUE=0x00000000 UPDATED_VALUE=0x00000001 REG_NAME="JOB_SLOT0_JS_COMMAND_NEXT"
HOST TIME=1515565849.691322s gpu.INFO_IrqJobControl: IRQ_ID=0x01 IRQ_NAME="JOB
Control" IRQ_STATE=Y
HOST TIME=1515565849.691561s gpu.INFO_ReadRegister: REG_OFFSET=0x000000000000100c
VALUE=0x00000001 REG_NAME="JOB_IRQ_STATUS"
HOST TIME=1515565849.691643s gpu.INFO_WriteRegister: REG_OFFSET=0x0000000000001004
VALUE=0x00000000 UPDATED_VALUE=0x00000001 REG_NAME="JOB_IRQ_CLEAR"
```

```
HOST_TIME=1515565849.691647s gpu.INFO_IrqJobControl: IRQ_ID=0x01 IRQ_NAME="JOB
Control" IRQ_STATE=N
```

4.11.3 Mapping between SYSREG_UPDATE trace sources and SPSR registers

For tracing updates to SPSR_* registers, GenericTrace maps the fields in the registers to fields in SYSREG_UPDATE32 or SYSREG_UPDATE64 trace sources.

The mapping is shown in the following table:

Table 4-10: Mapping between SYSREG_UPDATE* trace sources and register encodings for SPSR_* registers

SYSREG_UPDATE32 or SYSREG_UPDATE64 field	Register field
opc0	R
opc	M
CRn	M1
CRm	O
opc2	O

4.12 libete-plugin

The libete-plugin enables the Embedded Trace Extension (ETE) for AEMvA and applicable CPU implementations.



ETE support is integrated into the CT models. We recommend you configure ETE using the model parameters that are prefixed with `ete.`, which are equivalent to the plug-in parameters, instead of using this plug-in. The plug-in is deprecated and will be removed in a future release.

4.12.1 libete-plugin - parameters

This section describes the parameters for the libete-plugin plug-in.

Each parameter is prefixed with `TRACE.libete-plugin`, for example:

```
TRACE.libete-plugin.ASYNC_PACKETS_WHEN_VIEWINST_OFF
```


Table 4-11: libete-plugin parameters

Name	Type	Default value	Allowed values	When set	Description
ASYNC_PACKETS_WHEN_VIEWINST_OFF	bool	false	true, false	Init time	Generate the non-periodic alignment synchronisation packet generation when trace unit is operative.
ATBTRIG	bool	true	true, false	Init time	ATB trigger support.
CCITMIN	int	0x4	0x0 - 0xffff	Init time	Minimum cycle count value.
CCSIZE	int	0xc	0xc - 0x14	Init time	Cycle counter size.
CLAIMTAGS	int	0x8	0x0 - 0x20	Init time	Number of claim tags.
COMMOPT	bool	true	true, false	Init time	Commit mode.
COMMTRANS	bool	false	true, false	Init time	Commit transaction mode.
DEBUG	int	0x2	0x0 - 0xf	Runtime	DEBUG.
DESIGNER	int	0x41	0x0 - 0xff	Init time	DESIGNER value.
ETE_REVISION	int	0x0	0x0 - 0x3	Init time	ETE revision: 0=ETEv1.0, 1=ETEv1.1, 2=ETEv1.2, 3=ETEv1.3.
EXCEPTION_WITH_CONTEXT	bool	true	true, false	Init time	Whether EXCEPTION_WITH_CONTEXT packet is supported.
EXPLICITLY_COMMIT_PO_ELEMS	bool	false	true, false	Init time	Whether to unilaterally explicitly emit a commit after a PO packet.
IMPDEFEXCEPPERCENTAGE	int	0x0	0x0 - 0x32	Init time	Percentage of IMPDEF exceptions inserted in instruction blocks.
IMPDEF_TRACE_ON	int	0x0	0x0 - 0x3	Init time	Whether trace is flushed and trace on packet generated by events described by bitmap value. bit 0 - PE entering low power state, bit 1 - PE entering debug state.
IMPRECISE_FILTERING	int	0x0	0x0 - 0x14	Init time	Number of instruction blocks traced on a transition in the filtering.
LPOVERRIDE	bool	true	true, false	Init time	Low power override.
MAXSPEC	int	0x0	0x0 - 0xffffffff	Init time	Maximum speculation depth.
MAX_INST_PER_Q	int	0x1	0x1 - 0x1000	Init time	Maximum limit for the number of instructions implied by a Q element.
NOOVERFLOW	bool	false	true, false	Init time	No overflow.
NUMACPAIRS	int	0x4	0x0 - 0x8	Init time	Number of instruction address comparators pairs.
NUMCIDC	int	0x1	0x0 - 0x8	Init time	Number of context ID comparators.
NUMCNTR	int	0x2	0x0 - 0x4	Init time	Number of counters.
NUMEXTINSEL	int	0x4	0x0 - 0x4	Init time	Number of external input selectors.
NUMPC	int	0x0	0x0 - 0x8	Init time	Number of PE comparators.
NUMSEQSTATE	int	0x4	0x0 - 0x4	Init time	Number of sequencer states.
NUMSSCC	int	0x1	0x0 - 0x8	Init time	Number of single shot comparators.
NUMVMIDC	int	0x1	0x0 - 0x8	Init time	Number of virtual ID comparators.
NumberOfETEEEvents	int	0x2	0x0 - 0x4	Init time	Number of trace events.
NumberOfRSPairs	int	0x8	0x0 - 0x10	Init time	Number of resource selector pairs.
PIDR_CM0D	int	0x0	0x0 - 0xf	Init time	TRCPIDR CM0D value.
PIDR_DESIGNER	int	0x0	0x0 - 0x7ff	Init time	TRCPIDR DESIGNER value.

Name	Type	Default value	Allowed values	When set	Description
PIDR_PART	int	0x0	0x0 - 0xffff	Init time	TRCPIDR PART number value.
PIDR_REVAND	int	0x0	0x0 - 0xf	Init time	TRCPIDR REVAND value.
PIDR_REVISION	int	0x0	0x0 - 0xf	Init time	TRCPIDR REVISION value.
QFILT	bool	false	true, false	Init time	Q filtering.
QSUP	int	0x0	0x0 - 0x3	Init time	Q support.
Q_CADENCE	int	0x1	0x1 - 0x1000	Init time	Number of instruction blocks traced between two Q elements.
RAZWI_REG_SEL_TOP_BIT	bool	false	true, false	Init time	Implement Resource Selectors or Resource Selector Pairs bits as RAZ/WI.
REG_ACCESS_ONLY_MODE	bool	false	true, false	Init time	If enabled, all traces are disabled. Plugin only allows register acceses.
RES0_STATEFUL	bool	false	true, false	Init time	Whether RES0 bits are stateful or RAZ/WI.
RETSTACK	int	0x3	0x0 - 0xf	Init time	Return stack depth.
REVISION	int	0x0	0x0 - 0xf	Init time	TRCIDR1 revision value.
SIM_OVERFLOW_GRANULARITY	int	0x64	0xa - 0xffffffff	Init time	Number of instruction blocks in each granule, for simulated overflow.
SIM_OVERFLOW_PERCENTAGE	int	0x0	0x0 - 0x63	Init time	Percentage of instruction blocks lost in each granule, for simulated overflow.
SOURCE_ADDRESS	bool	false	true, false	Init time	Allow generation of source address elements.
STALLCTRL	bool	true	true, false	Init time	Stall control.
SYSSTALL	bool	true	true, false	Init time	System stall.
TRACEIDSIZE	int	0x7	0x0 - 0x7	Init time	Trace ID size.
TRACE_OUTPUT	string	""	""	Init time	File to which to write trace byte stream.
TRACE_OUTPUT_ENABLE	bool	false	true, false	Init time	ETE Trace output enable: 1=enable, 0=disable.
TRCRSRTA_FORCED_EXCEP	bool	false	true, false	Init time	TRCRSR.TA value for a forcibly traced exception.
TSMARK	bool	false	true, false	Init time	Whether timestamp markers are supported.
TSSIZE	int	0x8	0x8 - 0x8	Init time	Timestamp size.
WFXMODE	bool	true	true, false	Init time	WFX mode.

4.13 ListTraceSources

ListTraceSources is an MTI plug-in that displays a complete and self-documenting list of the trace sources that a model provides, without running the model.

The plug-in prints output for each component in the model, either to stdout or to a file. For example:

```
...
Component (4) providing trace: FVP_VE_Cortex_A7x1.cluster.cpu0 (ARM_Cortex-A7,
11.4.60)
=====
Component is of type "ARM_Cortex-A7"
```

```
Version is "11.4.60"
#Sources: 195

Source ASYNC MEMORY_FAULT (Context ID Register write.)
  Field FAULT type:MTI_UNSIGNED_INT size:4 (Fault status in ESR format)
  Field VADDR type:MTI_SIGNED_INT size:8 (Virtual Address (or 0 if unavailable))
  Field PADDR type:MTI_UNSIGNED_INT size:8 (Physical Address (or 0 if
  unavailable))
  ...
```

The source code for this plug-in is also provided as a programming example in `$PVLIB_HOME/examples/MTI/ListTraceSources/source/`.

4.13.1 ListTraceSources - parameters

This section describes the parameters for the ListTraceSources plug-in.

Each parameter is prefixed with `TRACE.ListTraceSources`, for example:

```
TRACE.ListTraceSources.file
```

Table 4-12: ListTraceSources parameters

Name	Type	Default value	Allowed values	When set	Description
file	string	""	""	Init time	File to write the list of trace sources to. Default is to write to the console.
print_components_only	bool	false	true, false	Init time	If true, the plug-in prints the trace component information only, not the sources or fields.

4.14 PipelineModel

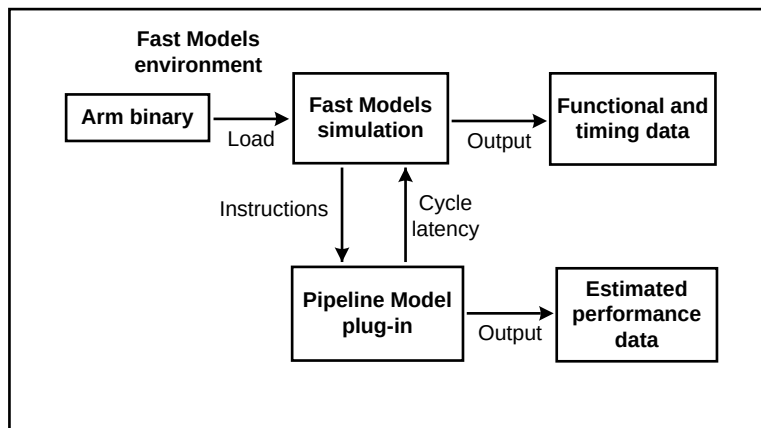
Use the `PipelineModel` plug-in to estimate the performance of workloads within a Fast Models environment.



The `PipelineModel` plug-in is deprecated. It might be modified or removed in a future release.

The plug-in models first-order effects of microarchitecture components on the overall Cycles Per Instruction (CPI) value. Examples of such effects are data and structural hazards due to instruction dependencies.

The `PipelineModel` is implemented as a Fast Models plug-in. It processes instruction traces and injects cycle latencies into the simulation. The plug-in is non-intrusive to the functional accuracy of the simulation.

Figure 4-1: PipelineModel plug-in overview

Fast Models provides the following prebuilt example PipelineModel plug-ins:

CortexA53PipelineModel

An approximation of the performance characteristics of the in-order, dual issue properties of the Cortex®-A53 processor. The model provides estimated performance characteristics of a given compute-bound workload in terms of Cycles Per Instruction.



The model should not be used as a reference for hardware performance as it has limitations, such as the absence of a cache model.

InOrderPipelineModel

An implementation of a single-issue 4-stage pipelined processor that illustrates the basic components of a CPU. It demonstrates how components such as Fetch, Decode, and Execute can be implemented as a Fast Models plug-in. The source code is provided in `$FVLIB_HOME/plugins/source/PipelineModel/Cores/InOrder/`. It contains a README and a makefile for building the example.



The model does not represent any Arm® core and is intended only as a guide for developing more advanced PipelineModels using Fast Models.

The `PipelineModel` is an aspect of Timing Annotation. For more details, see [Timing Annotation](#) in the Fast Models User Guide.

4.14.1 CortexA53PipelineModel - parameters

This section describes the parameters for the CortexA53PipelineModel plug-in.

Each parameter is prefixed with `PipelineModel.CortexA53PipelineModel`, for example:

```
PipelineModel.CortexA53PipelineModel.core-type
```

Table 4-13: CortexA53PipelineModel parameters

Name	Type	Default value	Allowed values	When set	Description
core-type	string	""	""	Init time	Core type description.
end-pc	string	"0x00000000"	""	Init time	End processing instructions after this pc.
instance-name	string	""	""	Init time	Cluster and core num description.
output	string	""	""	Init time	Defaults to stdout. Specify STDERR or a filename.
start-pc	string	"0x00000000"	""	Init time	Start processing instructions from this pc.

4.14.2 InOrderPipelineModel - parameters

This section describes the parameters for the InOrderPipelineModel plug-in.

Each parameter is prefixed with `PipelineModel.InOrderPipelineModel`, for example:

```
PipelineModel.InOrderPipelineModel.core-type
```

Table 4-14: InOrderPipelineModel parameters

Name	Type	Default value	Allowed values	When set	Description
core-type	string	""	""	Init time	Core type description.
instance-name	string	""	""	Init time	Cluster and core num description.

4.14.3 PipelineModel example

This example shows how the `PipelineModel` plug-in generates data. This example uses a single issue, 6-stage pipeline. The stages are Fetch, Decode, Issue, EX1, EX2, and WR.

		0	1	2	3	4	5	6	7	8	9	10
[0]	ADD R1, R2, R3	IF	ID	IS	X0	X1	WR					
[1]	MUL R4, R1, R2		IF	ID	IS	IS*	X0	X1	WR			
[2]	ADD R5, R4, R6			IF	ID	ID*	IS	IS*	X0	X1	WR	
[3]	ADD R7, R8, R9				IF	IF*	ID	ID*	IS	X0	X1	WR

In this instruction sequence, pipeline stalls are shown with an asterisk. The following stalls occur:

- Instruction [1] stalls one cycle in the Issue stage until R1 has been written.

- Instruction [2] stalls one cycle in the Decode stage due to a structural hazard because the Issuer is still in use by Instruction [1].
- Instruction [2] stalls one cycle in the Issue stage until R4 has been written.
- Instruction [3] stalls one cycle in the Fetch stage due to a structural hazard because the Decoder is still in use by Instruction [2].
- Instruction [3] stalls one cycle in the Decode stage due to a structural hazard because the Issuer is still in use by Instruction [2].

The `PipelineModel` plug-in uses the accumulated stalls for each instruction to calculate the runtime latency as follows:

- Instruction [0] latency = 0 cycles.
- Instruction [1] latency = 1 cycle.
- Instruction [2] latency = 2 cycles.
- Instruction [3] latency = 2 cycles.

In this example, there are five stall cycles in total. At the end of the simulation, the `PipelineModel` plug-in uses the accumulated stalls to produce a total cycle count. It uses the cycle count to determine the final Cycles Per Instruction (CPI) or Instructions Per Cycle (IPC) value.

4.14.4 Naming the plug-in instance

You can optionally assign a name to the plug-in instance. This is useful in a multiprocessor platform if you load the same plug-in multiple times. The assigned name is used to identify which plug-in instance the parameters apply to.

For example, the following commands load the `PipelineModel` plug-in twice and assign the names `cortexA53_0` and `cortexA53_1` to the first and second instances respectively:

```
--plugin CortexA53_0=CortexA53PipelineModel.so  
--plugin CortexA53_1=CortexA53PipelineModel.so
```

You can then specify the plug-in instance name in the plug-in parameters. For example:

```
-C PipelineModel.CortexA53_0.core-type=ARM_Cortex-A53  
-C PipelineModel.CortexA53_0.instance-name=Base.cluster0.cpu0  
  
-C PipelineModel.CortexA53_1.core-type=ARM_Cortex-A53  
-C PipelineModel.CortexA53_1.instance-name=Base.cluster0.cpu1
```

4.14.5 Example command lines

The following command lines show how to load the `PipelineModel` plug-in with a variety of platforms. Any parameters that are not relevant to the examples have been omitted.

Default usage, with no options

The following command line attaches the plug-in to any instance and sends output to `stdout`:

```
./EVS_Base_Cortex-A53x1.x
...
--plugin CortexA53PipelineModel.so
```

Single cluster, single processor platform

The following command line attaches the plug-in to `cluster0.cpu0`, which is a Cortex®-A53 processor. It outputs the file `stat.txt`:

```
./EVS_Base_Cortex-A53x1.x
...
--plugin CortexA53PipelineModel.so
-C PipelineModel.CortexA53PipelineModel.core-type=ARM_Cortex-A53
-C PipelineModel.CortexA53PipelineModel.instance-name=Base.cluster0.cpu0
-C PipelineModel.CortexA53PipelineModel.output=stat.txt
```

Single cluster, multiprocessor platform, default usage

The following command line attaches the plug-in to all Cortex®-A53 processors:

```
./EVS_Base_Cortex-A53x2.x
...
--plugin CortexA53PipelineModel.so
```

Single cluster, multiprocessor platform

The following command line loads two plug-in instances in a dual-processor platform. The first plug-in instance is named `cortexA53_0` and is attached to `cluster0.cpu0`. The second plug-in instance is named `cortexA53_1` and is attached to `cluster0.cpu1`:

```
./EVS_Base_Cortex-A53x2.x
...
--plugin CortexA53_0=CortexA53PipelineModel.so
--plugin CortexA53_1=CortexA53PipelineModel.so
-C PipelineModel.CortexA53_0.core-type=ARM_Cortex-A53
-C PipelineModel.CortexA53_0.instance-name=Base.cluster0.cpu0
-C PipelineModel.CortexA53_1.core-type=ARM_Cortex-A53
-C PipelineModel.CortexA53_1.instance-name=Base.cluster0.cpu1
```

Multiclustler, single processor platform

The following command line names the plug-in instance `core1` and attaches it to `cluster1.cpu0`:

```
./EVS_Base_Cortex-A73x1-A53x1.x
...
--plugin Core1=CortexA53PipelineModel.so
-C PipelineModel.Core1.core-type=ARM_Cortex-A53
```

```
-C PipelineModel.Core1.instance-name=Base.cluster1.cpu0
```

Plug-in core-type mismatch

The following command line specifies a Cortex®-A55 processor. This mismatches the platform, which is Cortex®-A53. The plug-in will fail to load:

```
./EVS_Base_Cortex-A53x1.x
...
--plugin CortexA53PipelineModel.so
-C PipelineModel.CortexA53PipelineModel.core-type=ARM_Cortex-A55
```

Plug-in instance-name mismatch

The following command line specifies an `instance-name` that does not exist. The plug-in will fail to load:

```
./EVS_Base_Cortex-A53x1.x
...
--plugin CortexA53PipelineModel.so
-C PipelineModel.CortexA53PipelineModel.instance-name=Base.cluster0.cpu2
```

Range processing

The following command line specifies an address range. The `PipelineModel` only processes instructions within this range:

```
-C PipelineModel.CortexA53PipelineModel.start-pc=0x80000001
-C PipelineModel.CortexA53PipelineModel.end-pc=0x8000FFFF
```

If the start and end of the range are identical, the `PipelineModel` processes all instructions. This is the same as not specifying a range at all:

```
-C PipelineModel.CortexA53PipelineModel.start-pc=0x80000001
-C PipelineModel.CortexA53PipelineModel.end-pc=0x80000001
```

Cycle limit

To run the `PipelineModel` with a cycle limit, use the model option `--cyclelimit`. When the cycle limit is reached, the simulation terminates and the output file is generated.

The following command line causes the `PipelineModel` to process all instructions until the specified cycle limit is reached:

```
./EVS_Base_Cortex-A53x1.x
...
--cyclelimit=30000000
--plugin CortexA53PipelineModel.so
```




The `--cyclelimit` option starts counting from the instruction at the beginning of the simulation, not from the instruction at `start-pc`. If the cycle limit has been reached and the `start-pc` has not yet been hit, the `PipelineModel` does not process any instructions.

4.14.6 PipelineModel output

At the end of the simulation, the `PipelineModel` generates a Cycles Per Instruction (CPI) value along with other performance data. You can use this for further analysis.

For example:

```
Elapsed time: 8 seconds
Instructions per second: 484186
Simulated CPU speed: 1.484186 MHz
CPU cycles: 5159984
RAW stalls: 3551018
Instructions issued: 3873492
Instructions retired: 3873491
Loads executed: 921403
Stores executed: 923306
IPC: 0.750679
CPI: 1.33213
```

The CPI value is the primary metric that measures the performance of workloads. It is calculated using the formula:

$$\text{CPI} = \text{cycles elapsed} / \text{instructions retired}$$

The lower the CPI, the better the performance. As a general indication, a CPI of 0.5 on a dual-issue, in-order or out-of-order processor means that an instruction takes 0.5 cycles to complete. In this case, the pipeline units are maximized and no latencies are generated.

Conversely, performance can be measured in Instructions Per Cycle (IPC), where:

$$\text{IPC} = \text{instructions retired} / \text{cycles elapsed}$$

The higher the IPC, the better the performance. As a general indication, an IPC of two on a dual issue, in-order or out-of-order processor means that on average, two instructions commit in each cycle.

4.15 RunTimeParameterTest

`RunTimeParameterTest` is an example MTI plug-in that demonstrates how to add new string, integer, and boolean parameters at runtime.

This plug-in is provided only as source code, in `$PVLIB_HOME/examples/MTI/RunTimeParameterTest/source/`.

4.16 ScalableVectorExtension

The `scalableVectorExtension` plug-in enables AEMvA models to support the Scalable Vector Extension (SVE) and SVE2.



Note

- Support for SVE and SVE2 is built into the CT models. This built-in support gives behavior that matches the TRM. If you use this plug-in with one of these models, the plug-in overrides the built-in behavior. This plug-in is deprecated and will be removed in a future release.
- SVE and SVE2 are optional extensions to the Arm® architecture. For information about SVE and SVE2, see [Introduction to SVE](#) and [Introducing SVE2](#).
- To enable or disable support for particular features in the plug-in, use the `has_*` parameters.

4.16.1 ScalableVectorExtension - parameters

This section describes the parameters for the `ScalableVectorExtension` plug-in.

Each parameter is prefixed with `sve.ScalableVectorExtension`, for example:

```
sve.ScalableVectorExtension.clear_constrained_lanes
```

Table 4-15: ScalableVectorExtension parameters

Name	Type	Default value	Allowed values	When set	Description
<code>clear_constrained_lanes</code>	int	0x0	0x0 - 0x2	Init time	When a constrained vector length increases, previously inaccessible bits are set to zero. Possible values are: 0=never, 1=always, 2=if the register was written to while the vector length was constrained.
<code>combine_movprfx_and_destructive</code>	bool	false	true, false	Init time	Attempt to combine the execution of MOVPRFX and the destructively-encoded instruction that follows it.
<code>disable_speculative_accesses</code>	bool	false	true, false	Init time	All speculative memory accesses behave as though faulting, without accessing memory.
<code>enable_at_reset</code>	bool	false	true, false	Init time	Start with system registers set up for Scalable Vector Extension use.
<code>ffr_16b_pattern_UNKNOWN</code>	int	0x0	0x0 - 0xffff	Init time	A specific 16-bit UNKNOWN value that is used by parameter <code>force_UNKNOWN_to_ffr</code> .

Name	Type	Default value	Allowed values	When set	Description
force_UNKNOWN_to_ffr	int	0x0	0x0 - 0x3	Init time	Governs behavior if WRFFR writes a non-monotonic value to FFR. Possible values are: 0 - Write non-canonical value to FFR, 1 - Overwrite FFR with a specific pattern of 16-bit UNKNOWN value. See ffr_16b_pattern_UNKNOWN, 2 - Clear all bits above first zero 3 - Set all bits after first one.
fp_exception_report_lowest	bool	false	true, false	Init time	If true, for multiple trapped FP exceptions, report the lowest lane in VECITR. Otherwise, report the highest.
fp_exception_set_tfv	bool	true	true, false	Init time	Set ESR_ELx.TFV during FP exception. Trapped exception flags are valid.
fp_exception_set_vecitr	bool	false	true, false	Init time	If true, set ESR_ELx.VECITR during FP exception. Otherwise, set RES0.
has_b16b16	int	0x0	0x0 - 0x1	Init time	Whether FEAT_SVE_B16B16 is implemented. Possible values are: 0 - Not implemented, 1 - Implemented if either FEAT_SVE2 or FEAT_SME2 is implemented.
has_sme	bool	false	true, false	Init time	Whether SME is implemented (FEAT_SME).
has_sme2	bool	false	true, false	Init time	Whether SME2 is implemented (FEAT_SME2).
has_sme_f16f16	int	0x0	0x0 - 0x1	Init time	Whether FEAT_SME_F16F16 is implemented. Possible values are: 0 - Not implemented, 1 - Implemented if FEAT_SME2 is implemented.
has_sme_f64f64	int	0x1	0x0 - 0x1	Init time	If SME is implemented, whether double-precision FMOPA and FMOPS are implemented.
has_sme_f8f16	int	0x1	0x0 - 0x1	Init time	If SME2 is implemented, whether FEAT_SME_F8F16 is implemented.
has_sme_f8f32	int	0x1	0x0 - 0x1	Init time	If SME2 is implemented, whether FEAT_SME_F8F32 is implemented.
has_sme_fa64	bool	false	true, false	Init time	Whether FEAT_SME_FA64 is implemented.
has_sme_i16i64	int	0x1	0x0 - 0x1	Init time	If SME is implemented, whether instructions that accumulate 16-bit integer outer products into 64-bit integer tiles are implemented.
has_sme_lutv2	bool	false	true, false	Init time	Whether FEAT_SME_LUTv2 is implemented.
has_sme_priority_control	bool	true	true, false	Init time	Whether SME Priority Control is implemented.
has_sve2	bool	false	true, false	Init time	Whether SVE2 is implemented (FEAT_SVE2).
has_sve2_aes	int	0x2	0x0 - 0x2	Init time	If SVE2 is implemented, whether AES instructions are implemented. Possible values are: 0 - not implemented, 1 - SVE2 AESE, AESD, AESMC, and AESIMC are implemented (FEAT_SVE_AES), 2 - Same as 1 but in addition SVE2 PMULLB and PMULLT with 64-bit source are implemented (FEAT_SVE_PMULL128).
has_sve2_bit_perm	bool	true	true, false	Init time	If SVE2 is implemented, whether BitPerm instructions are implemented (FEAT_SVE_BitPerm).

Name	Type	Default value	Allowed values	When set	Description
has_sve2_sha3	bool	true	true, false	Init time	If SVE2 is implemented, whether SHA3 instructions are implemented (FEAT_SVE_SHA3).
has_sve2_sm4	bool	true	true, false	Init time	If SVE2 is implemented, whether SM4 instructions are implemented (FEAT_SVE_SM4).
has_sve_bf16	bool	true	true, false	Init time	Whether SVE BFloat16 instructions are implemented.
has_sve_extended_bf16	int	0x2	0x0 - 0x2	Init time	Deprecated: to enable FEAT_EBF16, use CPU parameter has_ebf16. Whether Extended BFloat16 instructions are implemented. Possible values are: 0 - Disabled, 1 - Enabled if SME or SVE is implemented, 2 - Enabled if SME is implemented.
has_sve_mm_f32	bool	true	true, false	Init time	Whether the SVE FP32 Matrix Multiply instructions are implemented (FEAT_F32MM).
has_sve_mm_f64	bool	true	true, false	Init time	Whether the SVE FP64 Matrix Multiply instructions are implemented (FEAT_F64MM).
has_sve_mm_i8	bool	true	true, false	Init time	Whether the SVE Int8 Matrix Multiply instructions are implemented (FEAT_I8MM).
movprfx_unpredictable_behavior	int	0x0	0x0 - 0x3	Init time	Defines the behavior of MOVPRFX and the instruction it immediately precedes when the behavior is CONSTRAINED UNPREDICTABLE. Possible values are: 0 - UNDEF execution from MOVPRFX, 1 - MOVPRFX and second half of instruction executes as NOP, 2 - NOP MOVPRFX only, 3 - UNDEF execution from MOVPRFX unless otherwise trapped.
predicated_sp_align_check_behaviour	int	0x0	0x0 - 0x3	Init time	Governs behavior of SP alignment checking for predicated memory accesses. Possible values are: 0 - Always perform, 1 - Skip if governing predicate is 0, 2 - Skip for contiguous accesses if governing predicate is 0, 3 - Skip for gather/scatter accesses if governing predicate is 0.
relax_sme_watchpoint_matching_16	bool	false	true, false	Init time	Whether memory accesses through Z and P registers in Streaming Mode and all accesses through ZA match watchpoints rounded to 16-byte alignment.
relax_sve_watchpoint_matching_16	bool	false	true, false	Init time	If FEAT_DEBUGv8p9 is implemented, whether memory accesses through Z and P registers outside Streaming Mode match watchpoints rounded to 16-byte alignment.
sm_tag_checked	bool	true	true, false	Init time	Whether SME, SVE, and SIMD&FP load and store instructions executed when the PE is in Streaming SVE mode perform a Tag Check.
sme2_version	int	0x0	0x0 - 0x1	Init time	The version of SME2 if implemented. Possible values are: 0 - FEAT_SME2, 1 - FEAT_SME2p1.
sme_only	bool	false	true, false	Init time	If SME is implemented, whether SVE functionality is available only when SM=1.
sme_ssve_fp8_support_level	int	0x0	0x0 - 0x3	Init time	If FEAT_SME2 and FEAT_FP8 are implemented, whether FP8 operations are supported in Streaming Mode where not implemented outside Streaming Mode. Possible values are: 0 - No support above FEAT_FP8, 1 - FEAT_SSVE_F8FMA, 2 - FEAT_SSVE_F8DOT4, 3 - FEAT_SSVE_F8DOT2.
sme_veclens_implemented	int	0x7	0x1 - 0x1f	Init time	Which SME vector lengths are implemented. Represented as a bitfield where bit[n]==1 implies SME vector length of 128*2^n bits is implemented.

Name	Type	Default value	Allowed values	When set	Description
smidr_el1_implementer_val	int	0x41	0x0 - 0xff	Init time	The value of SMIDR_EL1.Implementer.
smidr_el1_revision_val	int	0x0	0x0 - 0xff	Init time	The value of SMIDR_EL1.Revision.
sve2_version	int	0x0	0x0 - 0x1	Init time	The version of SVE2 if implemented. Possible values are: 0 - FEAT_SVE2, 1 - FEAT_SVE2p1.
sve_dabt_far_behaviour	int	0x0	0x0 - 0x2	Init time	Whether the FAR reported on a Data Abort is imprecise. Possible values are: 0 - FAR Precise, 1 - FAR not Precise on abort due to an SVE contiguous vector load/store in Streaming SVE mode, or an SME load/store. , 2 - As per 1, but only for predicated SVE/SME instructions.
sve_wp_far_behaviour	int	0x0	0x0 - 0x3	Init time	FAR reporting behavior on a Watchpoint debug exception. Possible values are: 0 - FAR Precise, 1 - FAR not Precise on abort due to an SVE contiguous vector load/store in Streaming SVE mode, or an SME load/store, 2 - FAR not valid on abort due to an SVE contiguous vector load/store in Streaming SVE mode, or an SME load/store, 3 - As per 1, but only for predicated SVE/SME instructions.
trace_za_tilewise	bool	true	true, false	Init time	Whether tile-wise accesses to ZA are traced tile-wise rather than array-wise. Note: if false, column-wise accesses cause an event for every vector in the tile.
undef_invalid_combined_movprfx	bool	true	true, false	Init time	If a combined MOVPRFX is invalid, raise an UNDEF exception. Otherwise, NOP the second half.This parameter is deprecated.
unknown_value	int	-0x2152215221522153	0x0 - -0x1	Init time	Simulated value for a state that has an UNKNOWN value after reset.
veclen	int	0x8	0x2 - 0x20	Init time	SVE vector length in units of 64 bits.
z_reg_on_load_fault_behaviour	int	0x0	0x0 - 0x1	Init time	Governs the behavior of destination Z-registers in case of a load fault. Possible values are: 0 - Register becomes UNKNOWN, 1 - Register is preserved.
za_on_svl_increase_behaviour	int	0x0	0x0 - 0x1	Init time	Controls the state of the previously inaccessible portion of the ZA registers on SVL increase. Possible values are: 0 - Retain values, 1 - Zero ZA.
za_tag_checked	bool	true	true, false	Init time	Whether memory accesses due to SME LDR and STR instructions that access the SME ZA array perform a Tag Check.

4.17 Sidechannel

The Sidechannel plug-in allows communication between the software on the host and software on the target. It is no longer used.

4.18 TarmacText

TarmacText is an MTI plug-in that extracts the architectural execution trace, also known as Tarmac, of a processor. TarmacText extracts the trace in a textual form and saves it in a file.



TarmacText is deprecated. We recommend you only use it if you specifically require the TarmacText trace format. Otherwise, use the TarmacTrace plug-in instead.

The plug-in allows you to trace multiple components simultaneously, saving the generated traces in different files.

Enable trace generation by setting the `component` parameter to the required component name or space-separated names for multiple components. The default value of `component` is empty, which means the plug-in finds and traces all active processors.

Output filenames are composed of a common prefix, configurable with the `log` parameter, followed by the name of the component, and terminated with the extension `.log`. The default value of the prefix is `tarmac`.



The platform name is trimmed from the component name.

4.18.1 TarmacText - parameters

This section describes the parameters for the TarmacText plug-in.

Table 4-16: TarmacText parameters

Name	Type	Default value	Allowed values	When set	Description
<code>component</code>	string	<code>"*"</code>	<code>""</code>	Runtime	A space separated list of component to trace. Supports globbing (see man 7 glob). Does not restrict nor extend the list of components for which the ExecStep Iris EventSource will be published.
<code>evs</code>	string	<code>""</code>	<code>""</code>	Runtime	Filename to log the binary tarmac into. Supports the substitution pattern <code>@COMP@</code> .
<code>exec</code>	string	<code>""</code>	<code>""</code>	Runtime	Shell command to execute to process the binary tarmac on the fly. Supports the substitution pattern <code>@COMP@</code> . The command must accept on its standard input the tarmac event stream.
<code>flush</code>	numeric	<code>0x0</code>	<code>0x0 - 0x1</code>	Runtime	Whether to flush the output files specified by the <code>log</code> parameter as often as possible. Decrease the performances.
<code>log</code>	string	<code>"tarmac.@COMP@.log"</code>	<code>""</code>	Runtime	Filename to log the text tarmac into. Supports the substitution pattern <code>@COMP@</code> .

Name	Type	Default value	Allowed values	When set	Description
start	numeric	""	""	Runtime	The amount of cycles (as defined by the PERIODIC event) to wait for before starting to trace. This can be used to reduce the impact of the tracing on performances until the portion of interest is reached. Does not impact the ExecStep Iris EventSource.

4.19 TarmacTrace

TarmacTrace is an MTI plug-in that prints Tarmac trace to `stdout` or to a file. This section describes the format of the output.

The trace might include instructions executed, program flow, updates to registers, memory accesses made by Arm® cores in the simulation, and other information. Plug-in parameters control the amount and type of information that is traced.



This plug-in can be used with [ToggleMTIPlugin](#).

4.19.1 TarmacTrace - parameters

This section describes the parameters for the TarmacTrace plug-in.

Each parameter is prefixed with `TRACE.TarmacTrace`, for example:

```
TRACE.TarmacTrace.end-instruction-count
```

Table 4-17: TarmacTrace parameters

Name	Type	Default value	Allowed values	When set	Description
end-instruction-count	int	0x0	0x0 - 0x7fffffffffffffff	Init time	The instruction count when the tracing should be stopped. Default is to never stop tracing.
instruction-count-is-per-target	bool	true	true, false	Init time	If true (default) then the start-instruction-count and end-instruction-count parameters apply to individual targets separately. If false, all components start and stop tracing at once when the first component reaches the instruction count.
loadstore-display-width	int	0x8	0x0 - 0x40	Init time	Memory transactions can involve up to 64 bytes. For easier readability these can be broken up into multiple memory access records with a smaller number of bytes. 0 means do not break up any transaction.

Name	Type	Default value	Allowed values	When set	Description
quantum-size	int	0x2710	0x1 - 0x7fffffff	Init time	Set the default quantum size used to compute when the tracing should start and stop, in instructions. This is overridden by the CORE_INFO.QUANTUM_SIZE trace source field of the component, if present.
quiet	bool	false	true, false	Init time	Limit output to trace information.
start-instruction-count	int	0x0	0x0 - 0x7fffffffffffffff	Init time	The instruction count when the tracing should start. Default is to trace from the beginning.
trace-file	string	""	""	Runtime	Trace output file. The default is an empty string, which means stdout unless MTI_TARMAC_LOG is set. STDOUT means stdout. STDERR means stderr. Setting this parameter at runtime causes the current trace file to be flushed and closed and a new one to be opened. Writing at runtime, STDOUT, STDERR, and MTI_TARMAC_LOG are not supported when trace-file-per-comp=1.
trace-file-per-comp	bool	false	true, false	Init time	Write trace to multiple files.
trace-inst-stem	string	""	""	Init time	Base instance path to select a group of instances to trace.
trace_aarch64_vfp_full_width	bool	false	true, false	Init time	Trace a write to an S or D register in AArch64 as a write to the corresponding V register.
trace_atomic	bool	true	true, false	Init time	Trace memory update by atomic operation.
trace_branches	bool	false	true, false	Init time	Trace changes of the program flow like direct or indirect branches and exception returns.
trace_bte	bool	true	true, false	Init time	Trace opcode rejected by BTE.
trace_bus_accesses	bool	false	true, false	Init time	Trace bus accesses by the core. This includes accesses by the caches of the core. This considerably slows down the model.
trace_cache	bool	true	true, false	Init time	Trace cache fills and evictions.
trace_core_registers	bool	true	true, false	Init time	Trace the core registers R0-R14, the CPSR, and the SPSR registers.
trace_cp15	bool	true	true, false	Init time	Trace writes to CP15 registers.
trace_dap	bool	true	true, false	Init time	Trace accesses on the debug access port.
trace_ete	bool	true	true, false	Init time	Trace packets generated by the ETE.
trace_events	bool	true	true, false	Init time	Trace events, for example exceptions and mode changes.
trace_exception_reasons	bool	true	true, false	Init time	Trace INFO_EXCEPTION_REASONS (M-class only so far).
trace_generic_events	bool	false	true, false	Init time	Trace generic events.
trace_gicv3	bool	true	true, false	Init time	Trace GICv3 memory mapped accesses.
trace_gicv3_comms	int	0x0	0x0 - 0x7	Init time	Trace GICv3 communications between cores and distributor. Bitfield; 1 = trace CPU; 2 = trace RD0; 4 = trace internal.
trace_gicv3_its	bool	false	true, false	Init time	Trace GICv3 ITS command execution.

Name	Type	Default value	Allowed values	When set	Description
trace_gicv3_reads	bool	false	true, false	Init time	Trace GICv3 memory mapped reads.
trace_gpt	bool	true	true, false	Init time	Trace packets generated by the GPT.
trace_hacdbss	bool	true	true, false	Init time	Trace packets generated by the HACDBS.
trace_hdbss	bool	true	true, false	Init time	Trace packets generated by the HDBSS.
trace_instructions	bool	true	true, false	Init time	Trace instructions.
trace_loads_store_memtype	bool	false	true, false	Init time	Show memory type information for core loads and stores.
trace_loads_stores	bool	true	true, false	Init time	Trace loads and stores that are triggered by instructions. These might go into the memory subsystem, into a cache, or into a TCM. This considerably slows down the model.
trace_mask_s_regs	bool	false	true, false	Init time	Represent non-updated bytes as ---- in S-registers trace.
trace_memory	bool	false	true, false	Init time	Trace memory accesses just outside the core.
trace_mmu	bool	true	true, false	Init time	Trace mmu tablewalks and associated information.
trace_mpu_events	bool	false	true, false	Init time	Trace MPU events.
trace_spe	bool	true	true, false	Init time	Trace SPE data written to memory.
trace_tag_loads_stores	bool	true	true, false	Init time	Trace tag loads and stores that are triggered by MTE instructions. These might go into the memory subsystem, into a cache, or into a TCM. This considerably slows down the model.
trace_vfp	bool	true	true, false	Init time	Trace the VFP and Neon registers, including FPSCR and FPEXC.
unbuffered	bool	false	true, false	Init time	Trace events as they arrive and flush each fwrite. Prints IT even when IS should be printed.
updated-registers	bool	false	true, false	Init time	Trace the updated value of registers rather than the written value.
use_inst_end_for_inst_counter	bool	false	true, false	Init time	When using the instruction count as the timestamp, if true, increase the instruction count at INST_END instead of INST. When using the simulation time as the timestamp, this parameter has no effect.
use_instr_cnt_as_timestamp	bool	false	true, false	Init time	Use the instruction count as the timestamp instead of the simulation time.

4.19.2 TarmacTrace file format

This topic describes conventions used in the syntax definitions for each trace type.

In the syntax definition for each trace type:

[X|Y]

Indicates a choice between X and Y.

{X}

Indicates that X is optional or configuration-dependent.

This is the common address definition that is used in the trace command syntax:

```
<vaddr>{:<paddr><psecurity>}
```

where:

<vaddr>

is the virtual address in hexadecimal format. See the note after this list.

<paddr>

is the physical address of the instruction in hexadecimal. See the note after this list. <paddr> is only present if it is different to <vaddr>.

<psecurity>

either **_ns** if the security regime of the physical address is Non-secure, or not present if the regime is Secure.



For 64-bit addresses, the value is written as either:

- 8 hex digits, if the value can be represented in 32 bits.
- 16 hex digits otherwise.

This is the virtual regime definition that is used in the trace command syntax:

```
0x<vbase>[_NS] <el>{ vmid=<vmid>}{, nG asid=<asid>}
```

where:

0x<vbase>

is the virtual address in hexadecimal format.

_NS

if present, specifies that the address is Non-secure. If not present, the address is Secure.

<el>

is the translation regime that owns the mapping. One of:

- **EL1_n**, meaning the Non-secure EL1&O translation regime.
- **EL2_n**
- **EL1_s**
- **EL3_s**

<vmid>

if present, is the VMID for Non-secure, non-hyp regimes.

nG

if present, specifies that the virtual regime is non-global.

<asid>

if present, is the ASID, for non-global regimes.

4.19.3 Tarmac Trace output example

This example output from the Tarmac Trace plug-in shows various types of trace output, including instruction, memory access, register, translation table walk, and TLB traces.

```
...
89 clk IT (89) 80000164 f9000820 O EL1h_n : STR x0,[x1,#0x10]
89 clk MW8 80002010:000080002010_NS 00000000_80000705
90 clk IT (90) 80000168 d28080a0 O EL1h_n : MOV x0,#0x405
90 clk R X0 00000000000000405
...
98 clk IT (98) 80000188 d5181000 O EL1h_n : MSR SCTLR_EL1,x0
98 clk R SCTLR_EL1 00000000:00001005
98 clk TTW ITLB LPAE 1:1 000080002010 0000000080000705 : BLOCK ATTRIDX=1 NS=0 AP=0
SH=3 AF=1 nG=0 16E=0 PXN=0 XN=0 ADDR=0x0000000080000000
98 clk TLB FILL cpu0.UTLB 1G 0x80000000 NS EL1_n vmid=0:0x0080000000_NS Normal
InnerShareable Inner=WriteBackWriteAllocate Outer=WriteBackWriteAllocate xn=0 pxn=0
ContiguousHint=0 xs=0
...
```

This trace shows three instructions:

- The first instruction is an `STR`, which stores the 64-bit value from register `x0` to the address in `x1 + 10` byte offset:

```
89 clk IT (89) 80000164 f9000820 O EL1h_n : STR x0,[x1,#0x10]
89 clk MW8 80002010:000080002010_NS 00000000_80000705
```

In more detail:

- `IT` is a label that indicates the type of trace event described by the line. `IT` means instruction taken. To interpret the values in this line, see [Instruction trace](#). For example:
 - `89` means this is the 89th instruction.
 - `clk` means that the preceding number is an instruction count. If `ps` was displayed here instead, this would indicate the first value is a timestamp.
 - `0x80000164` is the address from which the instruction was fetched.
 - `0xf9000820` is the 32-bit opcode of the instruction.
 - `o` indicates the CPU execution state, in this case AArch64.
 - `EL1h_n` indicates the current Exception level and Security state.
 - The rest of the line following the colon is the assembly language representation of the instruction.
- `MW8` indicates an 8-byte memory write. To interpret the values in this line, see [Processor memory access trace](#). For example:

- `0x80002010` is the virtual address to which the data was written. The value after the colon is the corresponding physical address. In this example, they are the same. The `_ns` suffix indicates that it is Non-secure memory.
 - `0x00000000_80000705` is the value of the data written. Each group of 8 digits is separated using an underscore.
- The second instruction is a `mov`, which moves the value `0x405` into register `x0`:

```
90 clk IT (90) 80000168 d28080a0 0 EL1h_n : MOV x0,#0x405
90 clk R X0 00000000000000405
```

- `R` indicates a register trace. To interpret the values in this line, see [Register trace](#). For example:
 - `x0` is the name of the register being written to.
 - `0x405` is the new value of `x0`, which is the value that was moved by the `mov` instruction.
- The third instruction is an `MSR`, which writes the value `0x1005` from register `x0` to System register `SCTLR_EL1`.

Writing to bit 0 of `SCTLR_EL1` enables the MMU, so that all subsequent memory accesses will be done through the MMU.

The next memory access following this instruction is an instruction fetch (not shown), so a Translation Table Walk (TTW) is required to find its address.

Following the TTW, the Translation Lookaside Buffer (TLB) is updated with the new entry that caches some of the values resulting from the TTW, for example region size, base address, cachability and sharability. This appears as a TLB trace:

```
98 clk IT (98) 80000188 d5181000 0 EL1h_n : MSR SCTLR_EL1,x0
98 clk R SCTLR_EL1 00000000:00001005
98 clk TTW ITLB LPAE 1:1 000080002010 0000000080000705 : BLOCK ATTRIDX=1 NS=0
AP=0 SH=3 AF=1 nG=0 16E=0 PXN=0 XN=0 ADDR=0x0000000080000000
98 clk TLB FILL cpu0.UTLB 1G 0x80000000 NS EL1_n vmid=0:0x0080000000 NS Normal
InnerShareable Inner=WriteBackWriteAllöcate Outer=WriteBackWriteAllöcate xn=0
pxn=0 ContiguousHint=0 xs=0
```

- `TTW` indicates a translation table walk trace. To interpret the values in this line, see [Translation table walk trace](#). For example:
 - `ITLB` means instruction TLB.
 - `LPAE` means Large Physical Address Extension (LPAE)-format translation table entries.
 - `1:1` means Walk stage 1, Walk level 1.
 - `0x000080002010` is the page base address and the page attributes.
 - `0x0000000080000705` is the raw translation table entry.
 - Following the colon is the parsed result. In this case, the LPAE region descriptor.
 - `TLB` indicates a TLB trace. To interpret the values in this line, see [TLB trace](#). For example:
 - `FILL` means a TLB fill operation.

- `cpu0.UTLB` means the operation is taking place on a Unified TLB, which is shared for I-side and D-side accesses.
- `1G` means the TLB entry is for a 1GB page.
- `80000000_ns` means the entry has a page base address of `0x80000000` and is Non-Secure.
- `EL1_n` means the entry is for EL1.
- `vmid=0:0x0080000000_ns` means the entry is tagged with a specific VMID.
- `Normal InnerShareable` means the entry is tagged as Normal Inner-Sharable.
- `Inner` and `outer` are the inner and outer cache attributes for this entry.
- `xn=0` means the entry is tagged NOT Execute Never.
- `pxn=0` means the entry is tagged NOT Privileged Execute Never.
- `ContiguousHint=0` means the entry is not tagged as part of a set of contiguous entries that can be cached as one entry.
- `xs=0` means the page for this TLB entry is NOT XS, indicating either lack of support for FEAT_XS in the core, or it is non-XS memory.

4.19.4 Instruction trace

If enabled, this trace source generates one record for every instruction started.

The records (lines) of the instruction trace have this syntax:

```
<time> <scale> <cpu> [IT|IS] (<inst_id>) <addr> <opcode> [A|T|X|O] <mode>_<security> :
<disasm>
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for `<time>`. `ps` means simulation time, `clk` means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

[IT|IS]

IT

Instruction passed the condition code (taken).

IS

Instruction failed the condition code (skipped).

<inst_id>

Tick count of this processor, which is equivalent to the number of instructions that are executed, except for certain instructions like `WFI/WFE` (decimal value).

<addr>

Fetch source address for this instruction. Formatted according to the common address definition, see [TarmacTrace file format](#).

<opcode>

16-bit or 32-bit hexadecimal opcode of the instruction.

[A|T|X|O]

Instruction set:

A

A32

T

T32

X

T32EE

O

A64

<mode>

Processor execution mode.

AArch32 modes are *svc*, *irq*, *fiq*, *usr*, *mon*, *sys*, *abt*, *und*, *hyp*.

AArch64 modes are *EL3h*, *EL3t*, *EL2h*, *EL2t*, *EL1h*, *EL1t*, *EL0t*.

<security>

Processor security state (*s* or *ns*).

<disasm>

Disassembly of the instruction.

4.19.5 Program flow trace

If enabled, every executed branch instruction triggers this trace source, which is a more efficient way to reconstruct the program flow than by tracing every instruction.

Output syntax:

```
<time> <scale> {<cpu>} [FD|FI] (<inst_id>) <addr> <targ_addr> [A|T|X|O]
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for <time>. *ps* means simulation time, *c1k* means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

[FD|FI]

Program flow change by:

FD

A direct branch.

FI

An indirect branch.

<inst_id>

Tick count of this processor, which is equivalent to the number of instructions that are executed, except for certain instructions like `WFI/WFE` (decimal value).

<addr>

Fetch source address for this instruction. Formatted according to the common address definition, see [TarmacTrace file format](#).

<targ_addr>

Address (virtual) at which the execution continues. Formatted according to the common address definition.

[A|T|X|O]

Instruction set after the branch:

A

A32.

T

T32.

X

T32EE.

O

A64

4.19.6 Register trace

If enabled, this source traces all writes to the processor registers.

This trace source includes writes to core registers `R0` to `R14`, `x0` to `x30`, `CPSR`, and `SPSR`, VFP registers such as `S0` to `S31`, `D0` to `D31`, `FPSCR`, and `FPEXC`, and writes to system registers including `CP14`, `CP15`, and `GIC`. Banked registers are traced separately using the mode as a suffix to the register name, for example `r13` (current register `R13`) and `r13_mon` (banked register `R13`).

Output syntax:

```
<time> <scale> {<cpu>} R <register> <value>
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for <time>. `ps` means simulation time, `clk` means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

<register>

Register name. Banked core registers can have a mode appended to them with a single underscore. Banked `CP14/CP15` registers have `_s` or `_ns` appended to indicate access of either the Secure or Non-secure banked register.



In Arm®v8 and Arm®v9, when the register name is `cpsr`:

- In AArch64 state, `cpsr` is used to trace PSTATE changes. The bit format of <value> follows the `SPSR_ELx` AArch64 format.
- In AArch32 state, the bit format of <value> follows the CPSR format.

<value>

Hexadecimal value that is written to the register (64 bits maximum).

If the SVE plug-in is loaded in the model, there are additional registers in the program view. The output examples below show how these registers are traced when the value changes. These data values can be very large.

```
8463 clk cpu0 IT (8439) 000282c0:0000152282c0_NS 053fc01f O EL1h_n : SEL
z31.B,p0,z0.B,z31.B
8463 clk cpu0 R z31 00000000_00000000_00000000_00000000
```

`R` indicates a register write. `z0` to `z31` are the vector registers. The written data are hexadecimal digits, which are separated by an underscore every 32 bits. The length of the written data varies with the configuration, depending on the vector length.

```
9756 clk cpu0 IT (9732) 01000074:000011000074_NS 2518e3e0 O EL1t_n : PTRUE p0.B,ALL
9756 clk cpu0 R p0 ffff
```

`R` indicates a register write. `p0` to `p15` are the predicate registers. The written data are hexadecimal digits. If they are long enough to require one, the digits are separated by an underscore every 32 bits. The length of the written data varies with the configuration, depending on the vector length. Predicate registers contain 1 bit per byte of vector register length.

4.19.7 Cache maintenance trace

If enabled, traces all cache maintenance operations that the processor initiates.

Output syntax:

```
<time> <scale> <cpu> CACHE MAINTENANCE <side> <operation> <scope> <data> {<pagesize>
<memtype>}
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for <time>. A value of `ps` means simulation time, `clk` means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

<side>

Data or instruction cache.

<operation>

Clean, invalidate, or both.

<scope>

By MVA or set/way, to Point of Coherency or Point of Unification, Inner Sharable or not.

<data>

Data that is associated with the operation. If the operation is by MVA, formatted according to the common address definition, see [TarmacTrace file format](#), otherwise use raw hexadecimal.

<pagesize>

If the operation is by MVA, this element is the size of the memory region that is described by the TLB entry that contains the MVA.

<memtype>

If the operation is by MVA, this element is the type of memory in the TLB entry that contains the MVA.

4.19.8 Cache content trace

Traces the movement of data into and out of the cache.

Output syntax:

```
<time> <scale> <cpu> CACHE <id> LINE <line> <operation> 0x<paddr><ns>
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for <time>. A value of `ps` means simulation time, `clk` means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

<id>

Level and side, or system identifier, of the cache.

<line>

Identifier of this line uniquely within this cache, expressed in hexadecimal.

<operation>

Notification for this cache line. One of the following options:

ALLOC

(Processor caches) Line contains new read data.

INVAL

(Processor caches) Line contains no data.

DIRTY

(Processor caches) Line contains new write data.

CLEAN

(Processor caches) Write data is written back, still valid for reads.

FILL

(System caches) Line is filled.

EVICT

(System caches) Line is evicted due to space pressure.

CLEAN

(System caches) Line is cleaned due to maintenance operation.

INVAL

(System caches) Line is invalidated due to maintenance operation.

<paddr>

Cache line physical address in hexadecimal.

<ns>

Cacheline security. Blank for Secure regime, or `_ns` for Non-secure regime.

4.19.9 Translation table walk trace

If enabled, this source traces all translation table walks initiated by the processor.

Output syntax:

```
<time> <scale> <cpu> [TTW|TTU] <side> <format> <stage>:<level> <address> <data> :  
<result>
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for <time>. `ps` means simulation time, `clk` means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

[TTW|TTU]

Translation table walks (reads) or translation table update (writes).

<side>

Data or instruction TLB.

<format>

VMSA or LPAE format translation table entries.

<stage>

Walk stage, within the range 1-2.

<level>

Walk level, within the range 1-3.

<address>

Physical address of lookup in hexadecimal.

<data>

Raw translation table entry in hexadecimal.

<result>

Parsed result. One of the following options:

ABORTED

The memory access caused a synchronous abort and no data was returned.

FAULT

The data that was returned is not valid for this stage and level.

RESERVED

The data that was returned is not valid for this stage and level.

TABLE {<attr>=<value>}+

Pointer to the next level of lookup, in LPAE format.

BLOCK {<attr>=<value>}

LPAE region descriptor.

SUPERSECTION {<attr>=<value>}

VMSA region descriptor.

SECTION {<attr>=<value>}

VMSA region descriptor.

PAGETABLE {<attr>=<value>}

Pointer to the next level of lookup, in VMSA format.

LARGEPAGE {<attr>=<value>}

VMSA region descriptor.

SMALLPAGE {<attr>=<value>}

VMSA region descriptor.

4.19.10 Granule protection table walk trace

If enabled, this source traces Granule Protection Table (GPT) walks. This event is triggered by GPT lookups.

Output syntax:

```
<time> <scale> <cpu> GPTW [ISIDE|DSIDE] L<level> <address> <descaddr> : <data>
<result>
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for <time>. `ps` means simulation time, `clk` means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

[DSIDE|ISIDE]

Data or instruction TLB.

<level>

GPT fetch level, either 0 or 1.

<address>

Physical address of the lookup.

<descaddr>

Physical address of the GPT descriptor.

<data>

GPT data that was read.

<result>

Parsed result. One of the following values:

BLOCK TYPE=0x01 GPI=<gpi> PGS=<pgs>

GPT descriptor is a Block descriptor. <gpi> is the GPI value of the fetched GPT entry.
<pgs> is the physical granule size.

CONTIGUOUS TYPE=0x01 GPI=<gpi> CRS=<crs>

GPT descriptor is a Contiguous descriptor. <gpi> is the GPI value of the fetched GPT entry. <crs> is the contiguous region size.

TABLE TYPE=0x03 ADDR=<addr>

GPT descriptor is a Table descriptor. <addr> is the next-level table address in hexadecimal.

GRANULE TYPE=0x0f GPI=<gpi> PGS=<pgs>

GPT descriptor is a Granule descriptor. <gpi> is the GPI value of the fetched GPT entry. <pgs> is the physical granule size.

INVALID

GPT entry is invalid.

4.19.11 TLB trace

If enabled, this source traces TLB entries that are filled and evicted by the processor.

Output syntax:

```
<time> <scale> <cpu> [TLB|WALKCACHE] FILL <id> <size> <virtualregime>:<paddr>
{<mementype>} {<attr>=<value>}+
```

or

```
<time> <scale> <cpu> [TLB|WALKCACHE] EVICT <id> <size> <virtualregime>
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the TRACE.TarmacTrace.use_instr_cnt_as_timestamp parameter.

<scale>

Unit for <time>. ps means simulation time, cik means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

<id>

Identifies which TLB or walk cache to trace.

<size>

Size of the region being mapped.

<virtualregime>

Virtual address and regime of the region being mapped, formatted according to the common virtual regime definition.

<paddr>

Physical base address of mapped region, formatted according to the common address definition, see [TarmacTrace file format](#).

<memtype>

For TLB entries, the memory type of the result. One of the following options:

Device- [G|nG] [R|nR] [E|nE] { (<alias>) }

Device memory, where:

[G|nG]

Gathering or nongathering.

[R|nR]

Reordering or nonreordering.

[E|nE]

Early write acknowledgement or not.

<alias>

Device-nGnRnE was previously known as StronglyOrdered.

Normal [NonShareable|Shareable] Inner=<cachetype> Outer=<cachetype>

Normal memory, where:

[NonShareable|Shareable]

Shareability

<cachetype>

[NonCacheable|WriteBack|WriteThrough] {NonReadAllocate} {Non}
{WriteAllocate}

[NonCacheable|WriteBack|WriteThrough]

Cacheability

{NonReadAllocate}

For cacheable memory, Read allocate hint. (Read allocate is assumed if not specified.)

{Non}{WriteAllocate}

For cacheable memory, Write allocate hint.

4.19.12 Event trace

If enabled, traces exceptions, interrupts, and exception returns. In AArch64, it also traces changes to the SPSel and to the current exception level, by generating a CoreEvent_ModeChange.

Output syntax:

<time> <scale> {<cpu>} E <value> {<mode>} {<value1>} <number> <desc>

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for <time>. `ps` means simulation time, `clk` means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

<value>

A value that is associated with the event, formatted according to the common address definition, see [TarmacTrace file format](#).

<mode>

For mode change events only, the new mode being entered.

<value1>

Where available, the hexadecimal representation of a second value that is associated with the event.

<number>

Event number.

<desc>

Event name.

In the following table, the `CoreEvent_CURRENT_*` and `CoreEvent_LOWER_*` events cover all the ways in which exception entry can happen in AArch64 state. For example, `CoreEvent_CURRENT_SPx_SYNC` corresponds to a synchronous exception taken from Current Exception level with `SP_ELx`, $x > 0$.

`CoreEvent_LOWER_64_IRQ` corresponds to an IRQ or vIRQ taken from Lower Exception level, where the implemented level immediately lower than the target level is using AArch64.

Table 4-18: Supported values for `value`, `number`, and `desc`

Number	Event description	Value
0x00000001	<code>CoreEvent_Reset</code>	-
0x00000002	<code>CoreEvent_UndefinedInstr</code>	-
0x00000003	<code>CoreEvent_SWI</code>	SWI number
0x00000004	<code>CoreEvent_PrefetchAbort</code>	-
0x00000005	<code>CoreEvent_DataAbort</code>	-
0x00000007	<code>CoreEvent_IRQ</code>	-
0x00000008	<code>CoreEvent_FIQ</code>	-
0x0000000E	<code>CoreEvent_ImpDataAbort</code>	-
0x00000019	<code>CoreEvent_ModeChange</code>	New mode
0x00000080	<code>CoreEvent_CURRENT_SPO_SYNC</code>	-

Number	Event description	Value
0x00000081	CoreEvent_CURRENT_SPO_IRQ	-
0x00000082	CoreEvent_CURRENT_SPO_FIQ	-
0x00000083	CoreEvent_CURRENT_SPO_ABORT	-
0x00000084	CoreEvent_CURRENT_SPx_SYNC	-
0x00000085	CoreEvent_CURRENT_SPx_IRQ	-
0x00000086	CoreEvent_CURRENT_SPx_FIQ	-
0x00000087	CoreEvent_CURRENT_SPx_ABORT	-
0x00000088	CoreEvent_LOWER_64_SYNC	-
0x00000089	CoreEvent_LOWER_64_IRQ	-
0x0000008A	CoreEvent_LOWER_64_FIQ	-
0x0000008B	CoreEvent_LOWER_64_ABORT	-
0x0000008C	CoreEvent_LOWER_32_SYNC	-
0x0000008D	CoreEvent_LOWER_32_IRQ	-
0x0000008E	CoreEvent_LOWER_32_FIQ	-
0x0000008F	CoreEvent_LOWER_32_ABORT	-

4.19.13 Processor memory access trace

If enabled, this source traces processor data accesses.

Output syntax:

```
<time> <scale> {<cpu>} M<rw><sz><attrib> <addr><data>
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for <time>. `ps` means simulation time, `clk` means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

<rw>

R

Read access.

W

Write access.

<sz>

Size of the data transfer in bytes (1, 2, 4, 8).

<attrib>

Optional access attribute:

X

Exclusive access.

T

Translated (unprivileged) access.

L

Locked access (SWP, SWPB instructions).

CAS<suffix>

Compare and swap operation, where <suffix> is either *c* or *d*. *_cas_c* shows the value to compare and *_cas_d* shows the value that will be written to memory if the comparison matches.



Note

The value that is stored in memory as a result of a compare and swap operation is shown by an *MU* trace source.

<addr>

Virtual address that is used to access memory. Formatted according to the common address definition, see [TarmacTrace file format](#).

<data>

Hexadecimal value of data transferred. The data padding is according to the size of the transfer. Data of 64 bits or more contains an underscore (*_*) separator every eight characters (32 bits).

4.19.14 Processor memory update trace

If enabled, this source traces memory update accesses caused by atomic operations.

Output syntax:

```
<time> <scale> {<cpu>} MU<sz>_<atomic_op> <addr> <data>
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for <time>. *ps* means simulation time, *clk* means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

<sz>

Size of the data transfer in bytes (1, 2, 4, 8, 16).

<atomic_op>

Atomic operation performed on this memory address:

ADD

Atomic add operation.

BIC

Atomic bit clear operation.

CASc

Atomic compare and swap operation.

EOR

Atomic exclusive or operation.

ORR

Atomic bit set operation.

SMAX

Atomic signed max operation.

SMIN

Atomic signed min operation.

SWP

Atomic swap operation.

UMAX

Atomic unsigned max operation.

UMIN

Atomic unsigned min operation.

<addr>

Physical address that is used to access memory. Formatted according to the common address definition, see [TarmacTrace file format](#).

<data>

Hexadecimal value of data that is stored in memory as a result of the atomic operation. Data of 64 bits or more contains an underscore (_) separator every eight characters (32 bits).

4.19.15 Memory bus trace

If enabled, this source traces transactions that are initiated through the memory bus master port of the processor. These accesses use physical addresses.

Output syntax:

```
<time> <scale> {<cpu>} B<rw><sz><fd><lk><p><s> I<wrcbs> O<wrcbs> <master_id> <addr>
<data>
```

<time>

Either the simulation time timestamp or the instruction count, depending on the value of the `TRACE.TarmacTrace.use_instr_cnt_as_timestamp` parameter.

<scale>

Unit for <time>. `ps` means simulation time, `clk` means instruction count.

<cpu>

Processor, or other component, that gave the instruction.

<rw>**R**

Read access.

W

Write access.

<sz>

Size of the data transfer in bytes.

<fd>**I**

Opcode fetch.

D

Data load/store or an MMU access.

<lk>**L**

Locked access.

X

Exclusive access.

_, underscore

Normal access.

<p>**P**

Privileged access.

_, underscore

Normal access.

<s>**S**

Secure access.

N

Non-secure access.

I<wrcbs>

Inner cache attributes. See o<wrcbs>.

O<wrcbs>

Outer cache attributes:

<w>

W

Allocate on write.

_, underscore

No allocate on write.

<r>

R

Allocate on read.

_, underscore

No allocate on read.

<c>

C

Cacheable access.

_, underscore

Non-cacheable access.

B

Bufferable access.

_, underscore

Non-bufferable access.

<s>

S

Shareability access.

_, underscore

Non-shareability access.

<master_id>

Master ID of the transaction.

<addr>

Physical address that is used to access memory, in hexadecimal format.

<data>

Hexadecimal value of data transferred. The data padding is according to the size of the transfer. Byte ordering is from lowest to highest byte. This ordering means that for accesses in little endian mode, the data occurs mirrored compared to the register/memory access records.

4.20 ToggleMTIPlugin

ToggleMTIPlugin is an MTI plug-in that can be used to limit the generation of trace by another plug-in to specific areas of interest.

Generating trace output throughout a simulation session can reduce simulation speed and result in very large trace files. ToggleMTIPlugin helps to avoid these problems by enabling you to toggle trace generation during the simulation. Toggling trace means that if trace is on, it is turned off, and vice versa.

ToggleMTIPlugin can be used with the following plug-ins:

- ASTFplugin
- GenericTrace
- TarmacTrace

4.20.1 ToggleMTIPlugin - parameters

This section describes the parameters for the ToggleMTIPlugin plug-in.

Each parameter is prefixed with `TRACE.ToggleMTIPlugin`, for example:

```
TRACE.ToggleMTIPlugin.diagnostics
```

Table 4-19: ToggleMTIPlugin parameters

Name	Type	Default value	Allowed values	When set	Description
diagnostics	bool	false	true, false	Init time	Print diagnostics.
disable_mti_from_start	bool	false	true, false	Init time	Enable or disable MTI callbacks from start of simulation.
disable_mti_runtime	bool	false	true, false	Runtime	Enable or disable MTI callbacks at runtime.
hlt_imm16	int	0xf000	0x0 - 0xffff	Init time	16-bit integer used in HLT instruction meant to be used by this plugin.
use_hlt	bool	true	true, false	Init time	If true, use HLT #imm16 instruction to toggle MTI behavior.

4.20.2 How to use ToggleMTIPlugin

As with other plug-ins, load ToggleMTIPlugin using the `--plugin` command-line option when launching the model.



- When loading ToggleMTIPlugin and any other trace plug-ins using the `--plugin` option, ToggleMTIPlugin must be the last plug-in to be specified on the command line.
- We recommend you disable trace generation from the start of the simulation, using the plug-in parameter `disable_mti_from_start=1`, then enable it when execution reaches the region of interest.

There are two alternative ways to use ToggleMTIPlugin. You cannot use both in the same simulation session. Use the `use_hlt` plug-in parameter to control which one to use:

- `use_hlt = 1`

To use this method, set the `hlt_imm16` plug-in parameter to an integer value. The application will use this value as the operand in `HLT` instructions to toggle MTI callbacks.

You must also set the following parameters on the core model that is running the application:

enable_trace_special_hlt_imm16

Set to true to enable the parameter `trace_special_hlt_imm16`.

trace_special_hlt_imm16

Specifies the integer value that is used as the operand to `HLT` instructions to cause the usual `HLT` execution to be skipped. If the value matches the value specified in the `hlt_imm16` plug-in parameter, tracing is toggled.

- `use_hlt = 0`

To use this method, set the runtime plug-in parameter `disable_mti_runtime` during the simulation session to either true to disable tracing, or false to enable tracing. Changes to the `disable_mti_runtime` parameter are ignored unless `use_hlt` is zero.

To change `disable_mti_runtime` at runtime, use a debugger, for example Model Debugger or use the `iris.debug` Python module. The example Python script `$PVLIB_HOME/examples/pyIris/inst_count_trace_control.py`, demonstrates how to do this.

5. Fast Models examples

The following top-level example directories are installed under `$PVLIB_HOME/examples`.

Table 5-1: Fast Models examples directories

Directory name	Description
CADI	Example C++ applications that demonstrate how to use the CADI debug API. Note: CADI is deprecated and will be replaced by the Iris debug API.
LISA	LISA+ source code and project files for FVPs.
LISAPlus	Example LISA+ components that show how to capture and generate MTI trace, remap PVBUS transactions, and handle burst transactions.
MTI	MTI plug-in examples that show how to extract and use trace information from models. The following examples are also available as pre-built libraries under <code>\$PVLIB_HOME/plugins</code> : <ul style="list-style-type: none"> GenericCounter. GenericTrace. ListTraceSources.
SystemCExport	<ul style="list-style-type: none"> Source code and makefiles for EVS platform examples and SVPs. LISA+ source for bridges and EVS components. Header files required for exporting LISA+ protocols to SystemC.



Note

On Microsoft Windows, the Fast Models installer creates a copy of the examples in `%USERPROFILE%\ARM\FastModelsPortfolio_%FM-VERSION%\examples\`. This copy allows you to save configuration changes to the examples without needing Administrator permissions.

5.1 CADI examples

Example CADI clients that demonstrate how to use the CADI API to perform common debugger operations. These include instantiating a new simulation, connecting to a running simulation, accessing memory and registers, setting breakpoints, and registering callbacks for asynchronous feedback from the simulation.

Each of these examples has a `readme.txt` that describes the example, how to run it, and in some cases, gives the expected output. The following CADI examples are provided:

Table 5-2: CADI examples

Example	Description
breakpoints	Sets a code breakpoint at a specific address in the application that is running on the model.
cache_dump	Displays information about the caches in the running simulation.

Example	Description
<code>cadi_lib</code>	Implements a shared library that contains a CADI simulation. You can load the library into Model Shell and connect a CADI-compliant debugger to it.
<code>cadi_server</code>	Implements an executable that contains a CADI simulation. It loads the plug-in library <code>CADIIPCRemoteConnection.so</code> , which enables you to connect a CADI-compliant debugger to it. You can run the <code>connecting_running_target</code> CADI example to mimic this connection.
<code>connecting_instantiating_model</code>	Instantiates a simulation that is a shared library, prints a list of all targets in the simulation, and connects to one of them.
<code>connecting_remote_target</code>	Connects to a remote simulation that has been configured to allow remote connections. The IP address and listening port are passed as parameters to the CADI client.
<code>connecting_running_target</code>	Connects to a running CADI target.
<code>disassembly</code>	Prints the disassembly of the instruction at the PC address in the application that is running on the model.
<code>memory</code>	Displays memory space information. Reads and displays some values from memory in the application that is running on the model.
<code>mti</code>	Loads the TarmacTrace MTI plug-in into the application that is running on the model and displays some tarmac trace output.
<code>registers</code>	Reads and displays register information for a CPU target in the model and reads and writes a specific register.
<code>tlb_dump</code>	Connects to a model and displays the contents of the TLBs.

Related information

[Component Architecture Debug Interface User Guide](#)

5.2 LISA examples

LISA+ source and project files for FVPs.



Note

The LISA platform examples are Integrated SIMulators (ISIMs). For more information about building and running them, see [Build and run an FVP example](#).

The following LISA examples are provided:

Table 5-3: LISA examples

Example	Description
<code>BusComponents</code>	Example LISA+ components that demonstrate different ways of using the <code>PVBus</code> interface.
<code>Common</code>	FVP-specific LISA+ components that are common to different types of FVPs.
<code>CSS</code>	Source and project files for Reference Design FVPs. These FVPs model compute subsystems (CSS) that target specific market segments. Reference software stacks are available for them, see Arm Ecosystem FVPs for more information.

Example	Description
FVP_Base	Source and project files for Base Platform FVP examples. For information about the Base Platform, see Base Platform .
FVP_BaseR	Source and project files for BaseR Platform FVP examples.
FVP_Base_RevC	Source and project files for Base Platform RevC FVP examples. For information about the Base Platform RevC, see Base Platform RevC .
FVP_Coproc_Demo	Example implementation of the <code>Coprocessor</code> interface. Registers the coprocessor with a ARMCortexM33CT or ARMAEMv8MCT model. For more information, see CoprocBusProtocol protocol .
FVP_MPS2	Source and project files for MPS2-based example platforms. For information about the MPS2 platforms, see Microcontroller Prototyping System 2 .
FVP_MPS3	Source and project files for MPS3-based example platforms that support the Arm®Corstone™ SSE-300 Example Subsystem. For more information, see MPS2 Platform FVPs in the FVP Reference Guide and Arm Corstone SSE-300 Example Subsystem Technical Reference Manual .
FVP_VE	Source and project files for VE FVPs. For information about the VE platform, see Versatile Express Model .
VP_PChannel	Shows how to create power controllers to control the power state of the cores and cluster, using the <code>PChannel</code> protocol. For information about <code>PChannel</code> , see PChannel protocol .

5.3 Build and run an FVP example

The FVP examples are located under `$PVLIB_HOME/examples/LISA/.f ### Before you begin`
{.section}

About this task

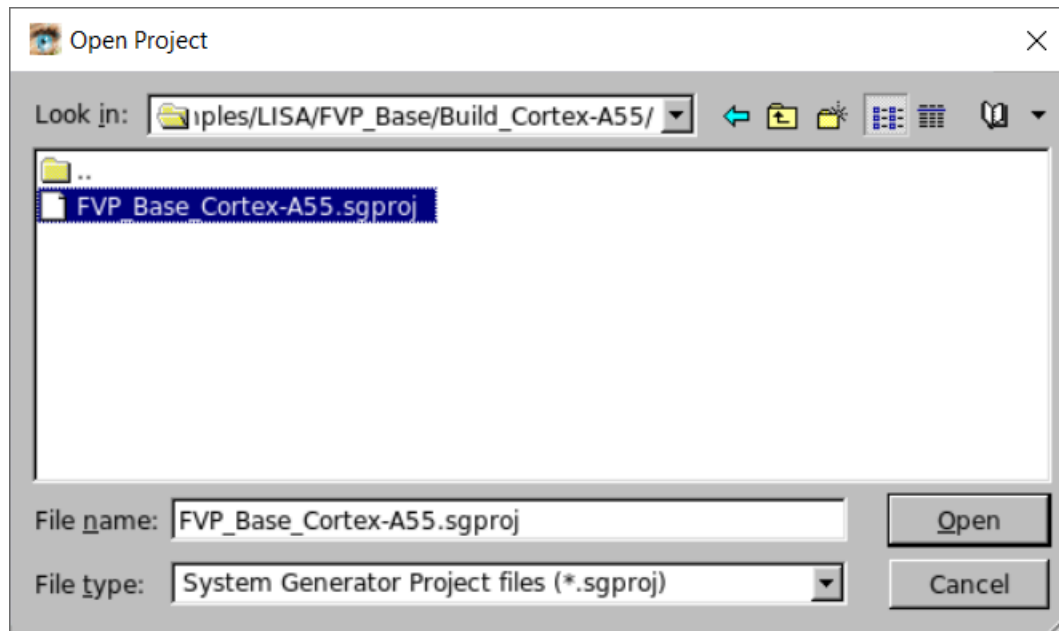
Follow these instructions to build and run one of the Base Platform FVP examples.



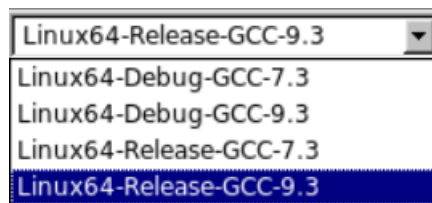
These examples build an executable platform model, which is sometimes referred to as an ISIM (Integrated SIMulator).

Procedure

1. These examples are built using System Canvas from a System Generator (SimGen) project file, with a `.sgproj` extension. To start System Canvas, open a terminal and type `sgcanvas`.
2. In System Canvas, select **File > Load Project...** to load the `.sgproj` file for the example you want to build. This example uses `$PVLIB_HOME/examples/LISA/FVP_Base/Build_Cortex-A55/FVP_Base_Cortex-A55.sgproj`.

Figure 5-1: System Canvas Open Project dialog box

- When the project has finished loading, select the build configuration from the Active Project Configuration drop-down menu on the main toolbar:

Figure 5-2: Select Active Project Configuration menu

- Click **Build** to build the FVP executable.
If you changed the active project configuration, click **Yes** when prompted to save the modified project file. The output from the build process is shown in the output window at the bottom of System Canvas. If the build is successful, the last message displayed is **Model Build process completed successfully**.
- The generated executable is named `isim_system`. In this example, it is created in `$PVLIB_HOME/examples/LISA/FVP_Base/Build_Cortex-A55/Linux64-Release-GCC-9.3`.
- You can run `isim_system` either from the terminal or from within System Canvas:
 - To run `isim_system` from the terminal:
 - Navigate to the directory where it is located.
 - To see a full list of command-line options for `isim_system`, run it with the `--help` option:

```
./isim_system --help
```

- The following example command-line shows how to load an application on `isim_system`:

```
./isim_system -a FVP_Base_Cortex_A55.cluster0.*=$PVLIB_HOME/images/  
image.axf -C bp.secure_memory=0
```

where:

-a

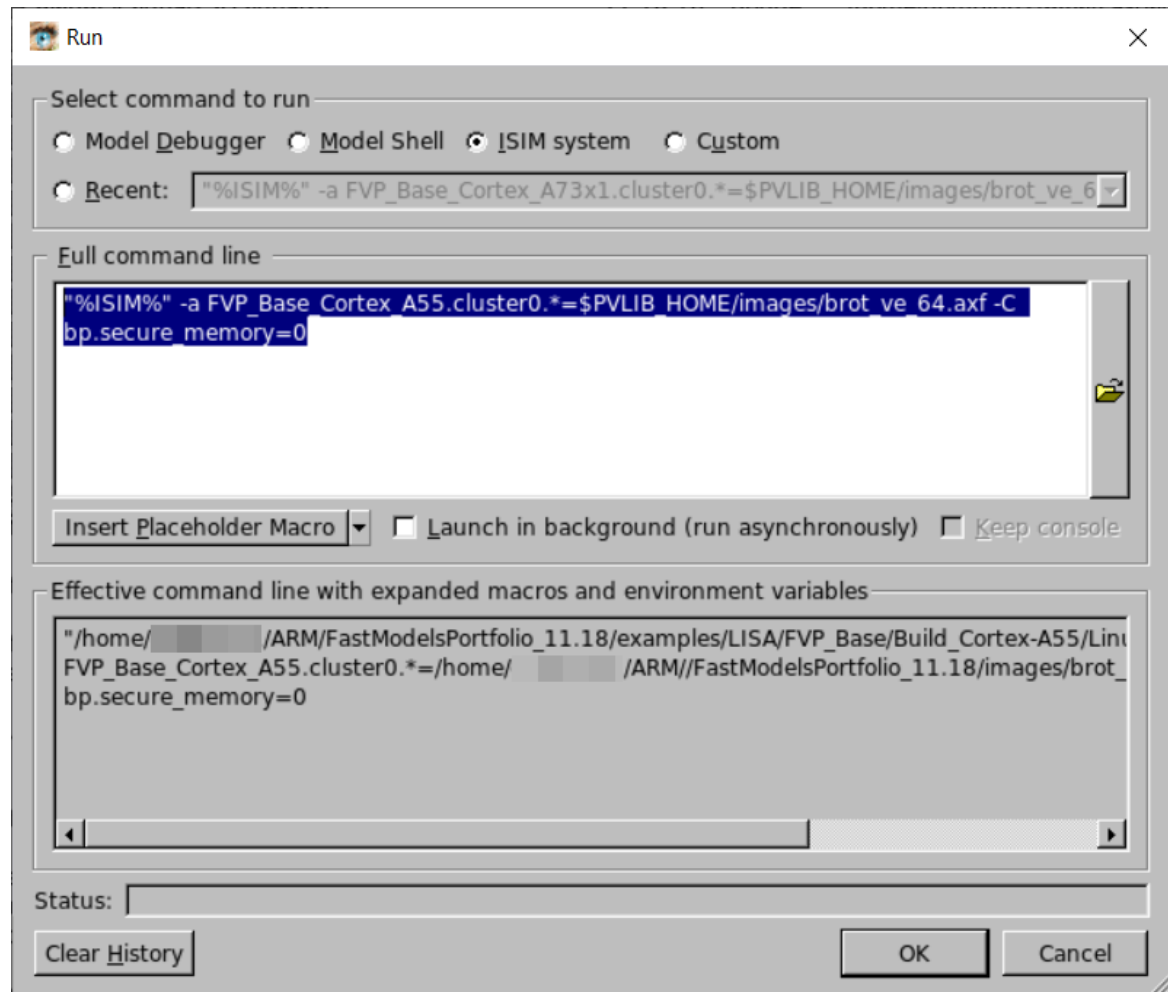
is the name of the application to load. Optionally it also specifies which core instances to run it on, in this case all cores in cluster0.

-C

is a configuration parameter. To see a full list of the available parameters, run the model with the `--list-params` option. To specify multiple parameters, it can be more convenient to place them in a text file, each parameter on a new line, and pass them to the model when starting it using `--config-file <filename>`.

- The ISIM starts running, displaying the output in a CLCD window.
- To exit the simulation, press **Ctrl+C**.
- To run `isim_system` in System Canvas:
 - Click Run.
 - In the Run dialog box, select the **ISIM system** checkbox, then enter the following command under **Full command line**:

```
"%ISIM%" -a FVP_Base_Cortex_A55.cluster0.*=$PVLIB_HOME/images/image.axf -C  
bp.secure_memory=0
```

Figure 5-3: System Canvas Run dialog box

- Click OK. The ISIM starts running, displaying the output in a CLCD window.
- To exit the simulation, click **Kill**.

Next steps

See the [System Canvas tutorial](#) in the Fast Models Tools User Guide for information on modifying, rebuilding, and debugging the example.

Related information

[System Canvas tutorial](#)

[Fast Models glossary](#)

[LISA examples](#) on page 4212

5.4 LISAPlus examples

Source code and System Canvas project files for some example LISA+ components.

Table 5-4: LISAPlus examples

Example	Description
CapturingTraceFromLISA	Instantiates and uses an MTI plug-in from a LISA+ component. Uses the SimpleTrace plug-in as an example.
GeneratingTraceFromLISA	Generates MTI trace information from a LISA+ component.
PVBusBursts	Uses a PVBusMaster to generate burst read transactions, which are handled by a PVBusSlave.
RemappingWithPVBusMapper	Uses the PVBusMapper component to remap transactions based on their attributes.

Related information

[PVBusMapper](#) on page 149

[PVBusMaster](#) on page 150

[PVBusSlave](#) on page 152

5.5 MTI examples

Example MTI plug-ins that show how to use MTI to extract and use trace information from models.

The following MTI examples are provided:

Table 5-5: MTI examples

Example	Description
CallTrace	Displays a function call sequence by tracing the PC field of INST trace sources, then compares the output with values in a symbol table. See the readme for more information.
CountingCacheStats	Registers counters for cache-related trace sources, for example CACHE_READ_HIT. Prints the cache stats before terminating.
DCCTrace	Prints the value of DBGDTRXX_ELO when data is written. Updates the TXFull bit in MDSCR to indicate the data was read. See the readme for more information.
GenericCounter	Registers a counter for trace sources. Prints the counter value for each INST trace source before terminating. This example is also available as a pre-built library, see GenericCounter .
GenericTrace	A flexible plug-in that traces one or more trace sources specified by the user. Prints the trace to a text file or to stdout. This example is also available as a pre-built library, see GenericTrace .
ITMtrace	Captures instrumentation trace macrocell (ITM) packets, which enables you to use ITM with a Cortex®-M class model. For more information about this plug-in, see Trace Cortex-M software with the Instrumentation Trace Macrocell (ITM) on Arm Community.

Example	Description
ListTraceSources	Displays either the trace sources provided by all trace components in the model, or just the trace components, to a text file or to stdout, without running the simulation. For more information, see <code>readme.txt</code> . This example is also available as a pre-built library, see ListTraceSources .
RunTimeParameterTest	Uses MTI to set runtime parameters.
SimpleTrace	Simple trace plug-in that prints a trace of the PC.
SoftwareTrigger	Traces SEMIHOSTING_PRECALL trace events, intercepts semihosting calls, and prints out register information. For more information, see the <code>readme</code> .
TraceOnBreak	Similar to the SimpleTrace example, but prints the PC value only when a breakpoint is hit.

Related information

[Fast Models Model Trace Interface Reference Manual](#)

5.6 SystemCExport examples

Components and platform models that are created by exporting LISA+ components or platforms to SystemC. Also, bridge components for converting transactions between LISA+ protocols and SystemC.

Table 5-6: SystemCExport examples

Directory	Description
Bridges	LISA+ source for bridge components.
Common	Source files and makefile rules that are common to the EVS and SVP examples.
Common/ Protocols	Header files that are required for the export of LISA+ protocols to SystemC.
EVS_Components	LISA+ files and project files for EVS (Exported Virtual Subsystem) components. These are LISA+ components with a SystemC wrapper and bridges that allow them to be used in a SystemC simulation.
EVS_Platforms	<p>LISA+ source and makefiles for EVS platform examples.</p> <p>An EVS platform is a LISA+ platform that has been exported as a SystemC object to allow it to be integrated into a SystemC simulation.</p> <p>The EVS platform examples are minimal platforms that are designed for a specific use case, for example running the Dhrystone benchmark application or booting Linux.</p> <p>The Dhrystone images <code>dhrystone_v8.axf</code> and <code>dhrystone.axf</code> for Armv7, are provided in the Fast Models Third Party IP package. They are loaded from <code>\$PVLIB_HOME/images/</code>.</p> <p>For more information about building EVS platforms, see Build and run an EVS platform example.</p>

Directory	Description
SVP_Platforms	<p>SVPs (SystemC Virtual Platforms) are platform models in which each component or subsystem has been individually exported to SystemC using the Fast Models Multiple Instantiation (MI) feature. For more information, see Building an SVP in the Fast Models User Guide.</p> <p>SVP platforms can be modified by replacing EVS components with other Fast Models EVSs, or with native SystemC components. There are three subdirectories:</p> <p>SVP_Base Armv8 Base Platform SVPs.</p> <p>SVP_bigLITTLE Armv7 big.LITTLE™ SVP, containing Cortex®-A15 and Cortex®-A7 clusters.</p> <p>SVP_LinuxBoot Minimal Armv7 platform that is suitable for booting Linux, containing a Cortex®-A15 cluster.</p>

5.7 Build and run an EVS platform example

The SystemCExport examples are located under `$PVLIB_HOME/examples/SystemCExport/`.

Before you begin

Ensure the following:

- You have installed Fast Models and SystemC 2.3.4 and have set the environment variables as described in [Installation](#) in the Fast Models User Guide.
- You have set up a Fast Models license, either using FlexNet Licensing or [User-Based Licensing](#).
- You are using a supported Operating System and have set up a compatible toolchain, from those listed in [Requirements for Fast Models](#) in the Fast Models User Guide.

About this task

Follow these instructions to build and run one of the EVS platform examples.



Note

These instructions also apply to the SVP platform examples, which are located in `$PVLIB_HOME/examples/SystemCExport/SVP_Platforms`.

Procedure

1. These examples are built using a Makefile. Open a terminal and navigate to the directory containing the example, and run `make`, specifying the build configuration, for example:

```
cd $PVLIB_HOME/examples/SystemCExport/EVS_Platforms/EVS_Base/Build_Cortex-A55
make rel_gcc93_64
```

This command creates the target executable `EVS_Base_Cortex-A55.x` and copies the shared objects that are required to run it into the current directory.

2. To see a full list of command-line options for the EVS platform, run it with the `--help` option:

```
./EVS_Base_Cortex-A55.x --help
```

The following example command-line shows how to load an application on the platform:

```
./EVS_Base_Cortex-A55.x \  
-a Base.cluster0.*=$PVLIB_HOME/images/image.axf \  
-C Base.bp.secure_memory=0
```

where:

-a

is the name of an application to load. Optionally it also specifies which core instances to run it on, in this case all cores in cluster0.

-C

is a configuration parameter. To see a full list of the available parameters, run the model with the `--list-params` option. To specify multiple parameters, it can be more convenient to place them in a text file, each parameter on a new line, and pass them to the model when starting it using `--config-file <filename>`.

3. The simulation starts running, displaying the output in a CLCD window.
4. To exit the simulation, press **Ctrl+C**.

Next steps

See the [System Canvas tutorial](#) in the Fast Models Tools User Guide for information on modifying, rebuilding, and debugging the example.

6. Base Platform

The Base Platform system model allows early development, distribution, and demonstration of software deliverables. A range of Base Platform FVPs are supplied as source code examples in Fast Models and as pre-built executables in the FVP Standard Library package.

For a list of the Base and BaseR Platform FVPs and the instances in them, see [Base Platform FVPs](#) and [BaseR Platform FVPs](#) in the FVP Reference Guide.

See `$PVLIB_HOME/examples/LISA/FVP_Base/` for LISA+ source and project files for the Base Platform FVP examples.

The standard peripheral set enables software development and porting. The platform is an evolution of the VE Fixed Virtual Platforms (FVPs), based on the Arm® Versatile Express (VE) hardware development platform.

It provides:

- Two configurable clusters of up to eight core models that implement:
 - AArch64 at all exception levels.
 - Configurable AArch32 support at all exception levels.
 - Configurable support for little and big endian at all exception levels.
 - Generic timers.
 - Self-hosted debug.
 - CADI debug.
 - GICv3 memory-mapped processor interfaces and distributor.
- Peripherals for multimedia or networking environments.
- Four PL011 UARTs.
- A CoreLink™ CCI-400 Cache Coherent Interconnect, or CCI-550 in Base Platform RevC.
- Architectural GICv3 model.
- High Definition LCD Display Controller, 1920*1080 resolution at 60fps, with single I2S and four stereo channels.
- 64MB NOR flash and board peripherals.
- CoreLink™ TZC-400 TrustZone® Address Space Controller.

Related information

- [LISA examples](#)

6.1 Base Platform RevC

Base Platform RevC is an evolution of the Base Platform, enhanced to support exploration of system-level virtualization.

Base Platform RevC has the following additions to the Base Platform:

- A PCIeRootComplex with these address regions:
 - A PCI-E config region at 0x0040000000 to 0x004ffffffffff. The PCI-E config region implements ECAM.
 - A PCI-E memory region at 0x0050000000 to 0x005ffffffffff.
 - A PCI-E memory region at 0x4000000000 to 0x7ffffffffff.
- An SMMUv3AEM with a control region at 0x002B400000 to 0x002B4FFFFFF.

The SMMUv3 is placed so that accesses to memory by PCI devices acting as bus masters are affected by it.

- A DMA330x4 with a control region at 0x002B500000 to 0x002B5FFFFFF.
- A Corelink™ CCI550 Cache Coherent Interconnect. Base Platform has a CCI-400.
- An AHCI controller model including a SATA disk model.
- An SMMUv3TestEngine component with a control region at 0x002bfe0000 to 0x002bffffffffff. This component is a traffic generator that acts as a (secure) device upstream of the SMMUv3.
- Two PCIe virtio devices are above the SMMU. By default they are configured to be device 0 and 2 on bus 0.
- The PCIe devices use a DeviceID that is the same as their RequestorID (BDF).
- Base Platform RevC AEMvA FVPs include a Mali_G76 GPU.
- Base Platform RevC AEMvA FVPs implement Armv8.0 by default, which does not support the Statistical Profiling Extension (SPE). To include SPE, add parameter `cluster0.has_arm_v8-3=1`, or similar, to the command line.

LISA+ source and project files for Base Platform RevC FVPs are located in `$PVLIB_HOME/examples/LISA/FVP_Base_RevC/`.

The FVP_Base_RevC-2xAEMvA model is freely available and does not require a license from Arm. You can download it from [Fixed Virtual Platforms](#) on Arm Developer.

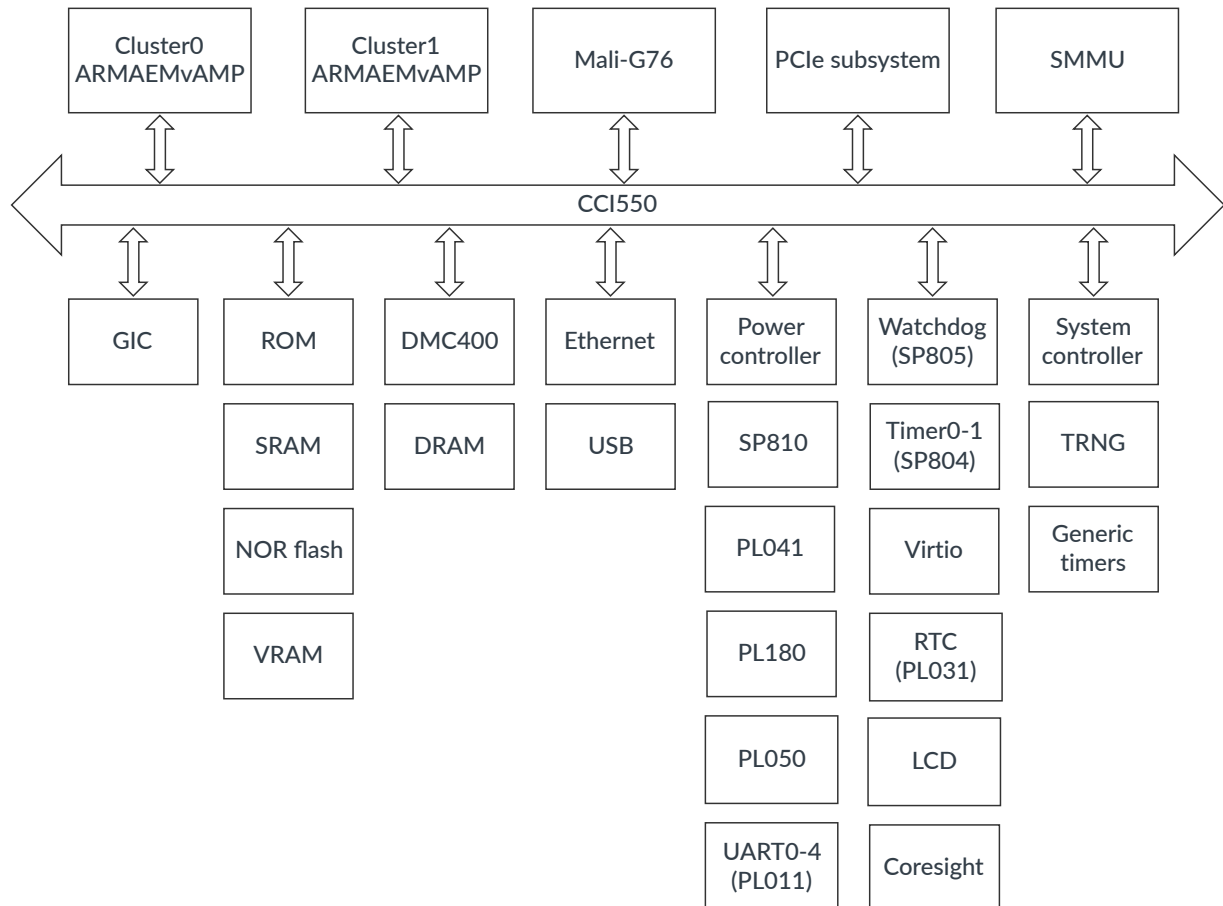
Related information

- [Base Platform memory map](#)
- [Base Platform interrupt assignments](#)

6.1.1 Base Platform RevC block diagram

This diagram shows the main components in Base Platform RevC. The platform shown is FVP_Base_RevC-2xAEMvA.

Figure 6-1: Base Platform RevC block diagram



6.1.2 BasePlatformPCIRRevC component

This component is an integrated PCIe subsystem which forms part of the Base Platform RevC. It incorporates an SMMUv3, a PCIe, an AHCI controller, and two PCI devices which wrap a pair of virtio PCI block devices. This model is written in LISA+.



Note

- You can include this component in a platform model, but Arm does not support using its subcomponents to create your own PCIe platform.
- The PCIe is not an implementation of any specific IP, but a functional, and limited, implementation of the PCIe standard.

BasePlatformPCIRRevC is composed of the following model components:

pci.pvbus2pci

The bridge from the Programmer's View bus to the PCI bus.

pci.pcidevice<n>

A wrapper around the underlying virtio block device. There are two block devices in the system, 0 and 1.

pci.pcivirtioblockdevice<n>

The instances of the virtio block device component.

pci.ahci_pci.ahci

An AHCI_SATA component with the following features:

- Each AHCI controller supports up to 32 simulated SATA disks. The configuration parameter `image_path` is a comma-separated list of one or more disk images.
- Supports 64-bit addresses.
- Supports plain, linear disk images, but also works with sparse files.

Some interesting options are:

- If the following options are set to non-zero values, they print messages about the operation of the bridge. The higher the value, the more verbose the component is:

```
pci.pvbus2pci.diagnostics=0x0 # (int) default = '0x0': Diagnostics level: [0x0..0x4]
pci.pcidevice<N>.diagnostics=0x0 # (int) default = '0x0': Diagnostics level:
[0x0..0x4]
```

- Each PCI device uses three BARs; one for config space, one for the MSI-X table structure and one for the MSI-X Pending Bit Array. Each of these can be configured to be 32 bits or 64 bits wide.

The Bus and Device number can be configured for each PCI device. If the device advertises MSI-X, support can be configured.

```
pci.pcidevice<N>.bus=0x0 # (int ) default = '0x0' : Bus number for this device :
[0x0..0xFF]
pci.pcidevice<N>.device=0x0 # (int ) default = '0x0' : Device number on this bus :
[0x0..0x1F]
pci.pcidevice<N>.bar0_64bit=0 # (bool) default = '0' : If BAR 0 is 64 bits wide, if
region size is nonzero
pci.pcidevice<N>.msix_support=0 # (bool) default = '0' : Enable device support for
MSI-X
pci.pcidevice<N>.bar2_64bit=0 # (bool) default = '0' : If BAR 2 is 64 bits wide, if
region size is nonzero
pci.pcidevice<N>.bar4_64bit=0 # (bool) default = '0' : If BAR 4 is 64 bits wide, if
region size is nonzero
```

- The following option configures the image file that the virtio block device exposes:

```
pci.pcivirtioblockdevice<N>.image_path="" # (string) default = '' : image file path
```

- There are two `PVBusLogger`s in the `pvbus2pci` component. One is in front of the Configuration space and one is in front of the Device space:

```
FVP_Base_AEMvA_AEMvA-PCI.pci.pvbus2pci.devicellogger
FVP_Base_AEMvA_AEMvA-PCI.pci.pvbus2pci.cfglogger
```

- There is one `PVBusLogger` in the `pcidevice` component. This reports on DMA accesses by the PCI device:

```
FVP_Base_AEMvA_AEMvA-PCI.pci.pcidevice0.dmalogger
```

- There is a `PVBusLogger` downstream of the SMMU. This reports on the transactions after they have been transformed by the SMMU:

```
FVP_Base_AEMvA_AEMvA-PCI.pci.smmulogger
```

Use the [GenericTrace](#) plug-in to capture traces from the loggers. For example, to see all accesses to device space, add the following options to the command line:

```
--plugin GenericTrace.so
-C TRACE.GenericTrace.trace-sources="FVP_Base_AEMvA_AEMvA-PCI.pci.pvbus2pci.devicellogger.*"
```

Use the [ListTraceSources](#) plug-in to list all the available trace sources in the model.

- To supply the AHCI controller with one or more SATA disk images, use the `image_path` parameter. For example:

```
-C pci.ahci_pci.ahci.image_path=disk1tb.img,disk8tb.img
```

Table 6-1: BasePlatformPCIRevC ports

Name	Protocol	Type	Description
<code>smmu_incoming_pvbus_s</code>	PVBus	Slave	Input port to smmu incoming device traffic slave based on PVBus protocol.
<code>pvbus_address_map_s</code>	PVBus	Slave	Input port to service transactions based on the PVBus protocol.
<code>pvbus_address_map_m</code>	PVBus	Master	Output port to send out PVBus transactions that are not handled by this component.
<code>system_reset</code>	Signal	Slave	Input port to handle reset signals. It is used to reset the internal state of this component.
<code>sev_out</code>	Signal	Peer	Port to send out a notification of the occurrence of an event as <code>sg::Signal</code> to a peer.
<code>interrupts[224]</code>	Signal	Master	Array of output ports of type <code>sg::Signal</code> to send out interrupts generated by this component.
<code>pvbus_pci_dma_m</code>	PVBus	Master	Output port to send out any DMA (of PVBus protocol) accesses originating from this component.
<code>clk_in</code>	ClockSignal	Slave	Input port to connect to a ClockSignal provider.

Table 6-2: BasePlatformPCIRRevC parameters

Name	Type	Allowed values	Default value	Description
ITS0-base	uint64_t	0 - 0xFFFFFFFFFFFFFFFF	0x2f020000	The ITS0 Base address.
pci_smmuv3. mmu.SMMU_IDR1	uint32_t	0 - 0xFFFFFFFF	0xe739d10	SMMU_IDR1.
pci_smmuv3. mmu.smmu_ msi_device_id	uint32_t	0 - 0xFFFFFFFF	0x10000	When appropriately enabled, assume that MSIs that are generated by the SMMU are presented to the GIC with this DeviceID. See parameters <code>msi_attribute_transform</code> and <code>enable_device_id_checks</code> .

Related information[SMMUv3AEM](#) on page 4046[VirtioBlockDevice](#) on page 2998

6.1.3 Base Platform RevC PCIeRootComplex

This component is an abstraction that represents the root of a PCIe device tree.

The addressable regions are also available as the following model parameters:

- PCIE_CFG_START, PCIE_CFG_END
- PCIE_MEM0_START, PCIE_MEM0_END
- PCIE_MEM1_START, PCIE_MEM1_END

PCIe device types:

Bridge

For example Root Port, Switch.

Endpoint

For example AHCI_SATA, SMMUv3TestEngine.

The PCIe device tree has the following properties:

- It always has PCIeRootComplex (RC) as its root.
- RC can connect to a Bridge or to an EndPoint.

PCIe device tree specification:

- The entire PCIe device tree can be specified through the model parameter `hierarchy_file_name` whose format is JSON. If no file is supplied, a default tree is assumed.

- Start the model and use the MTI trace source `ROOT_COMPLEX_HIERARCHY` to see the exact tree and parameter used for each device in the default tree. For example:

```
-C TRACE.GenericTrace.trace-  
sources="FVP_Base_RevC_2xAEMvA.pci.pcie_rc.ROOT_COMPLEX_HIERARCHY"
```

- The top-level devices of the default tree are: `AHCI_SATA`, `HostBridge`, `ROOTPORT0`, `ROOTPORT1`, `ROOTPORT2`, `ROOTPORT3`, `SMMUv3TestEngine0`, `SMMUv3TestEngine1`. `ROOTPORT1-4` are of type `PCieBridge` with parameter `port_type = 4`.

Error responses from PCIe devices:

- AMBA SLVERR (TX_ABORT) to PCIe Completer Abort (CA)
- AMBA DECERR (TX_DECODEABORT) to PCIe Unsupported Request (UR)

6.1.4 Base Platform RevC SMMUv3AEM

This is an architectural model implementing the SMMUv3.0 and SMMUv3.1 architectures which are for I/O virtualization of devices. The SMMU is placed so that accesses to memory by PCI devices acting as bus masters are affected by it.

The SMMU has the following features:

- Memory that is mapped to the range `0x2B400000-0x2B4FFFFFF`.
- Interrupts with IRQ IDs in the range 103-111.
- The event output pin of the SMMU is passed to the clusters.
- The downstream ports of the SMMU attach to the coherent bus infrastructure and so are coherent with the core clusters. All cores and the SMMU are in the same shareability domain. There is no distinction between the inner and outer shareability domains.
- The parameters of the SMMU determine its capabilities and have default values which can be overwritten if necessary.
- The SMMU is configured to only accept 16-bit StreamIDs and there is a 1:1 correspondence between RequestorID and StreamID.
- By default, the SMMU uses DeviceID `0x10000` to identify itself to the GIC (`pci.pci_smmuv3.mmu.smmu_msi_device_id`).

The parameter `gic_distributor.ITS-device-bits` is set to 17 by default to support the 17-bit DeviceIDs.

The SMMU has the following limitations:

- It does not support RAS.
- The PMU has limited functionality. Only a subset of the architecturally mandatory events are supported, as indicated by the `SMMU_PMC_CBEID0` fields. The PMU is intended for demonstration purposes only and for driver development.

6.1.5 Base Platform RevC PCIe device assignment

Every PCIe Endpoint can be enabled to support this feature which makes the Endpoint assignable to a Trusted Compute Base (TCB).

When an Endpoint is assigned to a TCB, it can be configured to both handle and generate memory transactions to specific memory regions.

The model has the following features:

- DOE read/write mailbox.
- SPDМ messages including FINISH_RESPONSE. There is no encryption of messages yet.
- IDE capability on device and root-port.
- Secure-SPDM. There is no MAC, encryption.
- TDISP messages and state machine.

To enable the device assignment feature in a PCIe RootPort, use the `rootportda_supported` parameter. To enable the feature in a PCIe Endpoint, use the following parameters:

doe_supported

Enables the PCIe extended capability called Data Object Exchange.

ide_supported

Enables the PCIe extended capability called Integrity and Data Encryption. This parameter also exists on the PCIe RootPort. The FVP does not model any encryption details but responds appropriately for “IDE” request messages.

x509_cert_der_filename

Filename containing an X-509 certificate issued for this DOE device in DER format. This certificate should contain the corresponding public key of the pair containing parameter `<rsa_priv_key_pem_filename>`.

rsa_priv_key_pem_filename

Filename containing a RSA private key, in PEM, for this DOE device. The corresponding RSA public key should be used to create parameter `<x509_cert_der_filename>`.

dmtf_meas_spec_info

Information about the DMTF measurement specification format. See the DMTF measurement specification format table for more information. Use the following format for the content of each line in this file:

`index_num_in_dec,dmtf_meas_spec_value_type,measurement_filename`

For example:

```
0,8,meas1.txt
1,9,meas2.txt
10,0,meas3.txt
```


6.1.6 Base Platform RevC legacy PCI interrupts

Each PCI device is hardwired to use INTA, with a value of 1 in the `interrupt_pin` register. This is required by the PCI specification for single-function devices.

The interrupts in the PCI host bridge are mapped according to section 2.2.6 of the PCI Local Bus Specification Revision 3.0, using the following formula, where the values for DeviceInterrupt are INTA = 0, INTB = 1, INTC = 2, INTD = 3:

$$\text{BridgeInterrupt} = (\text{Device} + \text{DeviceInterrupt}) \% 4$$

This formula produces the following mappings:

Table 6-3: Mapping BridgeInterrupts to DeviceInterrupts

BridgeInterrupt ID	DeviceInterrupt
200	INTA
201	INTB
202	INTC
203	INTD
207	SERR

6.1.7 Base Platform RevC MSI-X

The model optionally implements MSI-X, depending on whether the parameter `msix_support` is set. If set, an MSI-X capability is advertised as a PCI capability.



The Virtio specification is not fully compliant with the PCI specification and the virtio block device cannot be used in a 'pure polling' mode where MSI-X is always masked and only polling the Pending Bit Array is used.

The MSIs produced by the models, when directed to the GIC, have their payload rewritten to carry the DeviceID of the originating device to the GIC.

6.2 Base Platform startup configuration for v9 cores

Use the parameters `pctl.use_in_cluster_ppu` and `cluster.core_power_on_by_default` to configure startup of Arm®v9 cores in the Base Platform.

The following table shows the effect of each combination of these parameters.

Table 6-4: Configuring startup behavior for v9 cores in the Base Platform

Parameter configuration	Base Platform behavior	Recommended?
<code>pctl.use_in_cluster_ppu=true</code> <code>cluster.core_power_on_by_default=true</code>	All cores start up regardless of the PPU configuration done by the power controller. This configuration is invalid.	No
<code>pctl.use_in_cluster_ppu=false</code> <code>cluster.core_power_on_by_default=true</code>	Power controller loses the ability to do power state transitions altogether. All cores start running at once.	No
<code>pctl.use_in_cluster_ppu=true</code> <code>cluster.core_power_on_by_default=false</code>	<p>Only cores that are mentioned in the startup quad parameter are brought up.</p> <p>This configuration has the limitation that the RVBAR address must be supplied as a parameter on the command line.</p> <p>The application start address initialized by the application loader is overridden by the reset vector address provided in the parameter.</p>	Yes, but note the limitation described
<code>pctl.use_in_cluster_ppu=false</code> <code>cluster.core_power_on_by_default=false</code>	Wrong configuration. No core starts up.	No

6.3 Base Platform memory

This section describes the memory of the Base Platform.

6.3.1 Base Platform memory map

The Base Platform memory map is based on the Versatile™ Express RS2 memory map with extensions.



For an explanation of the values in the Security column, see [Base Platform secure memory](#).

Table 6-5: Base Platform memory map

Peripheral	Start address	Size	End address	Security
Trusted Boot ROM, secure flash, IntelStrataFlashJ3	0x00_0000_0000	64MB	0x00_03FF_FFFF	S
Trusted SRAM. 512KB is the default size. To set SRAM to 256KB instead, set the parameter <code>bp.secure_sram_size</code> to 0.	0x00_0400_0000	512KB	0x00_0407_FFFF	S
Trusted DRAM	0x00_0600_0000	32MB	0x00_07FF_FFFF	S

Peripheral	Start address	Size	End address	Security
NOR flash, flash0, IntelStrataFlashJ3	0x00_0800_0000	64MB	0x00_0BFF_FFFF	S/NS
NOR flash, flash1, IntelStrataFlashJ3	0x00_0C00_0000	64MB	0x00_0FFF_FFFF	S/NS
CS1-Pseudostatic RAM, PSRAM, on the motherboard.	0x00_1400_0000	64MB	0x00_17FF_FFFF	S/NS
VRAM	0x00_1800_0000	32MB	0x00_19FF_FFFF	S/NS
Ethernet, SMSC 91C111	0x00_1A00_0000	16MB	0x00_1AFF_FFFF	S/NS
USB, unimplemented	0x00_1B00_0000	16MB	0x00_1BFF_FFFF	S/NS
VE System Registers	0x00_1C01_0000	64KB	0x00_1C01_FFFF	S/NS
System Controller, SP810	0x00_1C02_0000	64KB	0x00_1C02_FFFF	S/NS
AACI, PL041	0x00_1C04_0000	64KB	0x00_1C04_FFFF	S/NS
MCI, PL180	0x00_1C05_0000	64KB	0x00_1C05_FFFF	S/NS
KMI - Keyboard, PL050	0x00_1C06_0000	64KB	0x00_1C06_FFFF	S/NS
KMI - Mouse, PL050	0x00_1C07_0000	64KB	0x00_1C07_FFFF	S/NS
UART0, PL011	0x00_1C09_0000	64KB	0x00_1C09_FFFF	S/NS
UART1, PL011	0x00_1C0A_0000	64KB	0x00_1C0A_FFFF	S/NS
UART2, PL011	0x00_1C0B_0000	64KB	0x00_1C0B_FFFF	S/NS
UART3, PL011	0x00_1C0C_0000	64KB	0x00_1C0C_FFFF	S/NS
Watchdog, SP805	0x00_1C0F_0000	64KB	0x00_1C0F_FFFF	S/NS
Base Platform Power Controller	0x00_1C10_0000	64KB	0x00_1C10_FFFF	S/NS
Dual-Timer 0, SP804	0x00_1C11_0000	64KB	0x00_1C11_FFFF	S/NS
Dual-Timer 1, SP804	0x00_1C12_0000	64KB	0x00_1C12_FFFF	S/NS
Virtio block device	0x00_1C13_0000	64KB	0x00_1C13_FFFF	S/NS
Virtio Plan 9 device	0x00_1C14_0000	64KB	0x00_1C14_FFFF	S/NS
Virtio net device	0x00_1C15_0000	64KB	0x00_1C15_FFFF	S/NS
Real-time Clock, PL031	0x00_1C17_0000	64KB	0x00_1C17_FFFF	S/NS
CF Card, unimplemented	0x00_1C1A_0000	64KB	0x00_1C1A_FFFF	S/NS
Color LCD Controller, PL111	0x00_1C1F_0000	64KB	0x00_1C1F_FFFF	S/NS
VirtioRNG entropy device	0x00_1C20_0000	64KB	0x00_1C20_FFFF	S/NS
LS64TestingFIFO	0x00_1D00_0000	64KB	0x00_1D00_FFFF	S/NS
Utility bus for DSU-110-enabled platforms	0x00_1E00_0000	16MB	0x00_1EFF_FFFF	NS
Non-trusted ROM, nontrustedrom	0x00_1F00_0000	4KB	0x00_1F00_0FFF	S/NS
CoreSight™ and peripherals	0x00_2000_0000	128MB	0x00_27FF_FFFF	S/NS
REFCLK CNTControl, Generic Timer	0x00_2A43_0000	4KB	0x00_2A43_0FFF	S
EL2 Generic Watchdog Control	0x00_2A44_0000	4KB	0x00_2A44_0FFF	S/NS
EL2 Generic Watchdog Refresh	0x00_2A45_0000	4KB	0x00_2A45_0FFF	S/NS
Trusted Watchdog, SP805	0x00_2A49_0000	64KB	0x00_2A49_FFFF	S
TrustZone® Address Space Controller, TZC-400	0x00_2A4A_0000	4KB	0x00_2A4A_0FFF	S
REFCLK CNTRead, Generic Timer	0x00_2A80_0000	4KB	0x00_2A80_0FFF	S/NS
AP_REFCLK CNTCTL, Generic Timer	0x00_2A81_0000	4KB	0x00_2A81_0FFF	S/NS
AP_REFCLK CNTBase0, Generic Timer	0x00_2A82_0000	4KB	0x00_2A82_0FFF	S
AP_REFCLK CNTBase1, Generic Timer	0x00_2A83_0000	4KB	0x00_2A83_0FFF	S/NS

Peripheral	Start address	Size	End address	Security
DMC-400 CFG, unimplemented	0x00_2B0A_0000	64KB	0x00_2B0A_FFFF	S/NS
SMMUv3 AEM (Base Platform RevC only)	0x00_2B40_0000	1MB	0x00_2B4F_FFFF	S/NS
DMA330x4 (Base Platform RevC only).	0x00_2B50_0000	1MB	0x00_2B5F_FFFF	S/NS
GIC Physical CPU interface, GICC. To configure the address of this region use model parameters. See [GICv3IRI].	0x00_2C00_0000	8KB	0x00_2C00_1FFF	S/NS
GIC Virtual Interface Control, GICH. To configure the address of this region use model parameters. See [GICv3IRI].	0x00_2C01_0000	8KB	0x00_2C01_1FFF	S/NS
GIC Virtual CPU Interface, GICV. To configure the address of this region use model parameters. See [GICv3IRI].	0x00_2C02_F000	8KB	0x00_2C03_0FFF	S/NS
CCI-400	0x00_2C09_0000	64KB	0x00_2C09_FFFF	S/NS
Mali™-G76 (Base Platform RevC only)	0x00_2D00_0000	16MB	0x00_2DFF_0000	S/NS
Non-trusted SRAM	0x00_2E00_0000	64KB	0x00_2E00_FFFF	S/NS
GICv3 IRI GICD. To configure the address of this region use model parameters. See [GICv3IRI].	0x00_2F00_0000	64KB	0x00_2F00_FFFF	S/NS
GICv3 IRI GITS. To configure the address of this region use model parameters. See [GICv3IRI].	0x00_2F02_0000	128KB	0x00_2F03_FFFF	S/NS
GICv3 IRI GICR. To configure the address of this region use model parameters. See [GICv3IRI].	0x00_2F10_0000	1MB	0x00_2F1F_FFFF	S/NS
PCIe config region (Base Platform RevC only)	0x00_4000_0000	256MB	0x00_4FFF_FFFF	S/NS
PCIe memory region 1 (Base Platform RevC only)	0x00_5000_0000	256MB	0x00_5FFF_FFFF	S/NS
Trusted Random Number Generator	0x00_7FE6_0000	4KB	0x00_7FE6_0FFF	S
Trusted Non-volatile counters	0x00_7FE7_0000	4KB	0x00_7FE7_0FFF	S
Trusted Root-Key Storage	0x00_7FE8_0000	4KB	0x00_7FE8_0FFF	S
DDR3 PHY, unimplemented	0x00_7FEF_0000	64KB	0x00_7FEF_FFFF	S/NS
HD LCD Controller, PL370	0x00_7FF6_0000	4KB	0x00_7FF6_0FFF	S/NS
DRAM. Memory Tagging Extension (MTE) is supported.	0x00_8000_0000	2GB	0x00_FFFF_FFFF	P
DRAM. Memory Tagging Extension (MTE) is supported.	0x08_8000_0000	30GB	0x0F_FFFF_FFFF	P
PCIe memory region 2 (Base Platform RevC only)	0x40_0000_0000	256GB	0x7F_FFFF_FFFF	S/NS
DRAM. Memory Tagging Extension (MTE) is supported.	0x88_0000_0000	480GB	0xFF_FFFF_FFFF	P
DRAM. Memory Tagging Extension (MTE) is supported.	0x00_0880_0000_0000	7.5TB	0x00_0FFF_FFFF_FFFF	P
DRAM. Memory Tagging Extension (MTE) is supported.	0x00_8800_0000_0000	120TB	0x00_FFFF_FFFF_FFFF	P
DRAM. Memory Tagging Extension (MTE) is supported.	0x08_8000_0000_0000	1920TB	0x0F_FFFF_FFFF_FFFF	P
DRAM. Memory Tagging Extension (MTE) is supported.	0x88_0000_0000_0000	2PB	0x8F_FFFF_FFFF_FFFF	P

Related information

[BaseR Platform memory map](#) on page 4233

6.3.2 Base Platform secure memory

Use the `bp.secure_memory` parameter to enable security checking on memory transactions by the TZC-400.

Table 6-6: Secure and Non-secure access permissions

Security	<code>bp.secure_memory=false</code>	<code>bp.secure_memory=true</code>
S	Secure and Non-secure access permitted.	Secure access is permitted, Non-secure access aborts.
S/NS	Secure and Non-secure access permitted.	Secure and Non-secure access permitted.
P	Secure and Non-secure access permitted.	Access conditions are programmable by the TZC-400.



The default state of the TZC-400 is to abort all accesses, even from Secure state.

Table 6-7: NSAIDs and filters that masters present to the TZC-400

Component	Non-Secure Access Identity (NSAID)	TZC-400 filter unit number (0-3)
Cluster 0	9	0
Cluster 1	9	0
VirtioNetMMIO	9	0
VirtioP9Device	8	0
VirtioBlockDevice	8	0
PL111_CLCD	7	2
HDLCD0	2	2

6.3.3 BaseR Platform memory map

The BaseR Platform copies its memory map from the Base Platform, but swaps the upper 2GB of address space with the lower 2GB.

Therefore:

- Any peripherals in the memory range `0x0` to `0x7FFFFFFF` in the Base Platform are available at the same offset in the memory range `0x80000000` to `0xFFFFFFFF` in the BaseR Platform.
- Memory in the range `0x80000000` to `0xFFFFFFFF` in the Base Platform is available at the same offset in the range `0x0` to `0x7FFFFFFF` in the BaseR Platform. For example, DRAM in the Base Platform memory map starts at address `0x80000000`. In the BaseR Platform, this is mapped to `0`.

The reason for this difference is that in the Arm®v8-R architecture, the upper 2GB of memory does not have execution permissions by default. So, code could not run from DRAM after reset if DRAM started at address `0x80000000`.

If the TCMs are enabled, for example with the parameter `-c cluster0.cpu0.tcm.a.enable=1`, then they are located at address 0. To move TCMs away from 0, use the parameters `itcm_base` and `dtcm_base`. For example:

```
-C cluster0.cpu0.itcm_base=0x8000 -C cluster0.cpu0.dtcm_base=0x18000
```

Related information

[Base Platform memory map](#) on page 4230

6.3.4 Base Platform DRAM

The multiple DRAM regions do not alias each other and form a contiguous 4PB area. The total amount of DRAM on the Base Platform system model is configurable. This ability affects where usable DRAM appears.

If the Base Platform system model has `bp.dram_size=4`, the default, then 2GB of DRAM is accessible at `0x00_8000_0000` to `0x00_FFFF_FFFF`, and the remaining 2GB is accessible at `0x08_8000_0000` to `0x08_FFFF_FFFF`.

If, instead, the Base Platform system model has `bp.dram_size=8`, then 2GB of DRAM is accessible at `0x00_8000_0000` to `0x00_FFFF_FFFF` and the remaining 6GB is accessible at `0x08_8000_0000` to `0x09_FFFF_FFFF`.

The default contents of RAM not otherwise written by the simulation is a repeating sequence of the following 64-bit value: `CFDFDFDFDFDFDFCF`.



Memory is allocated on demand, and performance degrades if very large amounts of memory are used.

6.4 Base Platform interrupt assignments

The platform assigns the Shared Peripheral Interrupts (SPIs) and Private Peripheral Interrupts (PPIs) on the GIC.



- SPI and PPI numbers are mapped onto GIC interrupt IDs as described in the Arm Generic Interrupt Controller Specification.
 - IRQ IDs 103-111, 192-194, and 200-207 apply to the Base Platform RevC only.
-

Table 6-8: SPI GIC assignments

IRQ ID	SPI offset	Device
32	0	Watchdog, SP805.
34	2	Dual-Timer 0, SP804.
35	3	Dual-Timer 1, SP804.
36	4	Real-time Clock, PL031.
37	5	UART0, PL011.
38	6	UART1, PL011.
39	7	UART2, PL011.
40	8	UART3, PL011.
41	9	MCI, PL180, MCIINTRO.
42	10	MCI, PL180, MCIINTR1.
43	11	AACI, PL041.
44	12	KMI - Keyboard, PL050.
45	13	KMI - Mouse, PL050.
46	14	Color LCD Controller, PL111.
47	15	Ethernet, SMSC 91C111.
56	24	Trusted Watchdog, SP085.
57	25	AP_REFCLK, Generic Timer, CNTPSIRQ.
58	26	AP_REFCLK, Generic Timer, CNTPSIRQ1.
59	27	EL2 Generic Watchdog WS0.
60	28	EL2 Generic Watchdog WS1.
74	42	Virtio block device.
75	43	Virtio P9 device.
76	44	Virtio net device.
78	46	VirtioRNG entropy device.
80	48	TZC-400.
92	60	cluster0.cpu0 PMUIRQ.
93	61	cluster0.cpu1 PMUIRQ.
94	62	cluster0.cpu2 PMUIRQ.
95	63	cluster0.cpu3 PMUIRQ.
96	64	cluster1.cpu0 PMUIRQ.
97	65	cluster1.cpu1 PMUIRQ.
98	66	cluster1.cpu2 PMUIRQ.
99	67	cluster1.cpu3 PMUIRQ.
103	71	SMMUv3 non-secure combined interrupt. Base Platform RevC only.
104	72	SMMUv3 secure combined interrupt. Unused because there is no secure side. Base Platform RevC only.
105	73	SMMUv3 secure event queue. Unused because there is no secure side. Base Platform RevC only.
106	74	SMMUv3 non-secure event queue. Base Platform RevC only.
107	75	SMMUv3 PRI queue. Unused because no PCIe device supports PRI. Base Platform RevC only.
108	76	SMMUv3 secure command queue sync. Unused because there is no secure side. Base Platform RevC only.

IRQ ID	SPI offset	Device
109	77	SMMUv3 non-secure command queue sync. Base Platform RevC only.
110	78	SMMUv3 secure GERROR. Unused because there is no secure side. Base Platform RevC only.
111	79	SMMUv3 non-secure GERROR. Base Platform RevC only.
117	85	HD LCD Controller, PL370.
139	107	Trusted Random Number Generator.
192	160	Mali™-G76 GPU. Base Platform RevC only.
193	161	Mali™-G76 GPU job. Base Platform RevC only.
194	162	Mali™-G76 GPU MMU. Base Platform RevC only.
200	168	PCIe INTA. Base Platform RevC only.
201	169	PCIe INTB. Base Platform RevC only.
202	170	PCIe INTC. Base Platform RevC only.
203	171	PCIe INTD. Base Platform RevC only.
207	175	PCIe SERR. Base Platform RevC only.

Table 6-9: PPI GIC assignments

IRQ ID	PPI offset	Device
19	3	Secure hypervisor virtual timer interrupt
20	4	Secure hypervisor physical timer interrupt
22	6	DCC, comms channel, interrupt
23	7	PMU, performance counter, overflow
24	8	CTI, Cross Trigger Interface, interrupt
25	9	Virtual CPU interface maintenance interrupt
26	10	Hypervisor timer interrupt
27	11	Virtual timer interrupt
28	12	Hypervisor virtual timer interrupt
29	13	Secure physical timer interrupt
30	14	Non-secure physical timer interrupt

6.5 Base Platform clocks

This table shows the clock frequencies of the Base Platform peripherals.

Table 6-10: Peripheral clock frequencies in the Base Platform

Device	Clock
Clusters	100MHz
REFCLK CNTControl, Generic Timer	100MHz
AP_REFCLK CNTCTL, Generic Timer	100MHz
Dual-Timer 0-1, SP804	35MHz
VE system registers	24MHz
UART 0-3, PL011	24MHz

Device	Clock
KMI 0-1, PL050	24MHz
MCI, PL180	24MHz
AACI, PL041	24MHz
Ethernet, SMSC 91C111	24MHz
Watchdog, SP805	24MHz
Color LCD Controller, PL111	23.75MHz
HD LCD Controller, PL370	10MHz
Trusted Watchdog, SP805	32.768kHz
Real-time Clock, PL031	1Hz

6.6 Base Platform parameters

This section describes the parameters.

Table 6-11: Base Platform parameters

Parameter	Type	Allowed values	Default value	Description
<code>bp.dram_size</code>	int	2, 4, or 8-4000000	4	Size of main memory in gigabytes: 2, 4, or any value between 8 and 4000000.
<code>bp.proc_id<n></code> . Some platforms do not expose this parameter.	uint32_t	-	Platform-specific	Processor ID for VE_SysRegs SYS_PROCID<n>.
<code>bp.secure_memory</code>	bool	true, false	true	The security state of the processor limits access to peripherals and RAM. See Note below table.
<code>bp.variant</code> . Some platforms do not expose this parameter.	uint32_t	0x0-0xF	Platform-specific	Board variant for VE_SysRegs SYS_ID.
<code>cache_state_modelled</code>	bool	true, false	true	Enable d-cache and i-cache state for all components.

When loading an image on an EVS, you might see the following message:



Note

```
Warning: Base.cluster0.cpu0: Uncaught exception, thread terminated
In file: gen/scx_scheduler_mapping.cpp:523
In process: Base.thread_p_5 @ 0 s
```

This warning means that the image is attempting to run from DRAM, which is access-controlled by the TZC_400 component. To disable security checking by the TZC_400, specify `-c Base.bp.secure_memory=false` when running the EVS.

Table 6-12: Base Platform debug parameters

Parameter	Type	Allowed values	Default value	Description
<code>dbgen</code>	bool	true, false	true	Debug authentication signal, dbgen.
<code>niden</code>	bool	true, false	true	Debug authentication signal, niden.

Parameter	Type	Allowed values	Default value	Description
spiden	bool	true, false	true	Debug authentication signal, spiden.
spniden	bool	true, false	true	Debug authentication signal, spniden.

6.7 Base Platform components

These component models implement some of the functionality of the Versatile Express (VE) hardware.

A complete model implementation of a Base Platform system model includes both Base Platform-specific components and generic components such as buses and timers. To see a list of all the component instances in the model, run it with the `--list-instances` option.

6.7.1 Base Platform Base_PowerController component

This section describes the Base_PowerController component.

6.7.1.1 Base_PowerController control interface

The Base Platform Power Controller component provides a basic register interface for software to control the power-up and power-down of cores in the cluster.

Identify cores in the system to the Base_PowerController by writing 24 bits in MPIDR format, providing the following levels of affinity:

Bits [23:16]

Affinity level 2.

Bits [15:8]

Affinity level 1.

Bits [7:0]

Affinity level 0.

Examples of affinity usage are `not_applicable/cluster/processor` and `cluster/processor/thread`.

To identify which cores to power up at startup, use parameter `pctl.startup`.

Specify core affinities with a dotted-quad. Wildcards are allowed. The format depends on the architecture:

- In Armv8.1 and earlier, use:

```
-C pctl.startup=0.0.Y.X
```

where X refers to the core number and Y refers to the cluster number. For example 0.0.0.0 refers to cluster0.cpu0 and 0.0.1.1 refers to cluster1.cpu1. Use wildcards to indicate all cores at an affinity level. For example, to turn on all the cores in cluster 0, use 0.0.0.*.

- In Armv8.2 and later, use:

```
-C pctl.startup=0.Z.Y.X
```

where X refers to the thread number, Y refers to the core number, and Z refers to the cluster number.

6.7.1.2 Base_PowerController registers

This section describes the registers.

Register summary

This section describes the power control registers in order of offset from the base memory address.

Table 6-13: Base_PowerController register summary

Offset	Name	Type	Reset	Width	Description
0x00	PPOFFR	RW	0x---	32	Power Control Processor Off Register
0x04	PPONR	RW	0x---	32	Power Control Processor On Register
0x08	PCOFFR	RW	0x---	32	Power Control Cluster Off Register
0x0C	PWKUPR	RW	0x---	32	Power Control Wakeup Register
0x10	PSYSR	RW	0x---	32	Power Control SYS Status Register

PPOFFR

The Power Control Processor Off Register (PPOFFR) characteristics are: purpose, usage constraints, configurations, and attributes.

Purpose

Processor SUSPEND command when PWKUPR and the GIC are programmed appropriately to provide wakeup events from IRQ and FIQ events to that processor.

Usage constraints

Processor must make power-off requests only for itself.

Configurations

Available in all configurations.

Attributes

See the register summary table.

Figure 6-2: Power Control Processor Off Register bit assignments

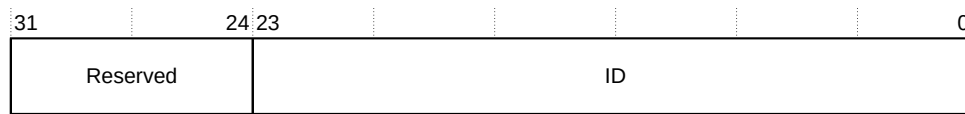


Table 6-14: Power Control Processor Off Register bit assignments

Bits	Name	Function
[31:24]	-	Reserved.
[23:0]	ID	MPIDR format affinity value of the processor to be switched off. Programming error if MPIDR != self.

PPONR

The Power Control Processor On Register (PPONR) characteristics are: purpose, usage constraints, configurations, and attributes.

Purpose

Brings up a processor from low-power mode.

Usage constraints

Processor must make power-on requests only for other powered-off processors in the system.

Configurations

Available in all configurations.

Attributes

See the register summary table.

Figure 6-3: Power Control Processor On Register bit assignments

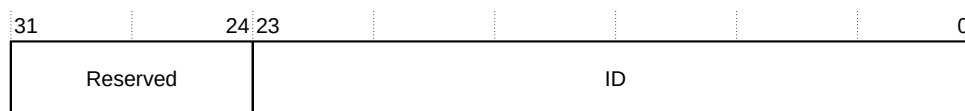


Table 6-15: Power Control Processor On Register bit assignments

Bits	Name	Function
[31:24]	-	Reserved.
[23:0]	ID	MPIDR format affinity value of the processor to be switched on. Programming error if MPIDR == self.

PCOFFR

The Power Control Cluster Off Register (PCOFFR) characteristics are: purpose, usage constraints, configurations, and attributes.

Purpose

Turns the cluster off.

Usage constraints

Cluster must make power-off requests only for itself.

Configurations

Available in all configurations.

Attributes

See the register summary table.

Figure 6-4: Power Control Cluster Off Register bit assignments

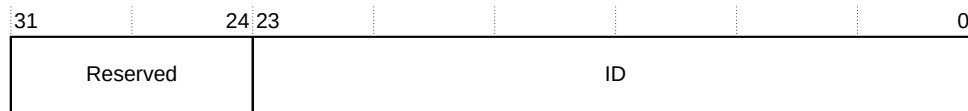


Table 6-16: Power Control Cluster Off Register bit assignments

Bits	Name	Function
[31:24]	-	Reserved.
[23:0]	ID	MPIDR format affinity value of powered-on processor in the cluster to be switched off. Programming error if MPIDR != self.

PWKUPR

The Power Control Wakeup Register (PWKUPR) characteristics are: purpose, usage constraints, configurations, and attributes.

Purpose

Configures whether wakeup requests from the GIC are enabled for this cluster.

Usage constraints

There are no usage constraints.

Configurations

Available in all configurations.

Attributes

See the register summary table.

Figure 6-5: Power Control Wakeup Register bit assignments

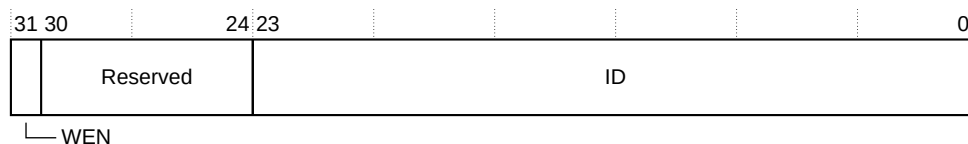


Table 6-17: Power Control Wakeup Register bit assignments

Bits	Name	Function
[31]	WEN	If set, enables wakeup interrupts (return from SUSPEND) for this cluster.
[30:24]	-	Reserved.
[23:0]	ID	MPIDR format affinity value of processor whose Wakeup Enable bit is to be configured.

PSYSR

The Power Control SYS Status Register (PSYSR) characteristics are: purpose, usage constraints, configurations, and attributes.

Purpose

Provides information on the powered status of a given core. Software writes bits [23:0] for the required core and reads the value along with the associated status in bits [31:24].

Usage constraints

There are no usage constraints.

Configurations

Available in all configurations.

Attributes

See the register summary table.

Figure 6-6: Power Control SYS Status Register bit assignments

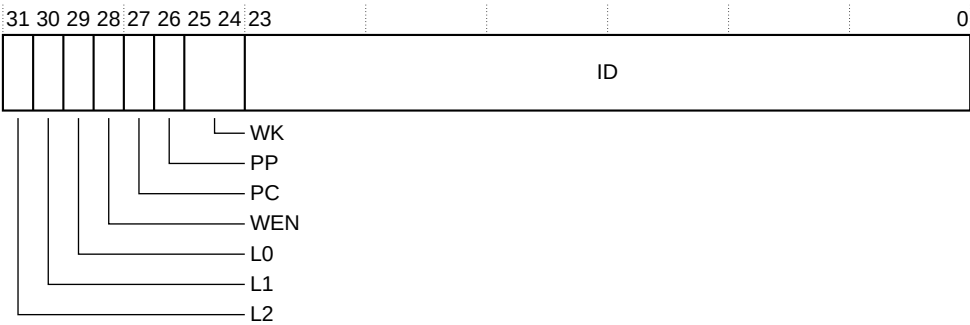


Table 6-18: Power Control SYS Status Register bit assignments

Bits	Name	Function
[31]	L2	Read-only. A value of 1 indicates that affinity level 2 is active/on. If affinity level 2 is not implemented this bit is RAZ .
[30]	L1	Read-only. A value of 1 indicates that affinity level 1 is active/on. If affinity level 1 is not implemented this bit is RAZ .
[29]	L0	Read-only. A value of 1 indicates that affinity level 0 is active/on.
[28]	WEN	Read-only. A value of 1 indicates wakeup interrupts, return from SUSPEND, enabled for this processor. This is an alias of PWKUPR.WEN for this core.
[27]	PC	Read-only. A value of 1 indicates pending cluster off, the cluster enters low-power mode the next time it raises signal STANDBYWFIL2.

Bits	Name	Function
[26]	PP	Read-only. A value of 1 indicates pending processor off, the processor enters low-power mode the next time it raises signal STANDBYWFI.
[25:24]	WK	Read-only. Indicates the reason for LEVEL0 power on: 0b00 Cold power-on. 0b01 System reset pin. 0b10 Wake by PPONR. 0b11 Wake by GIC WakeRequest signal.
[23:0]	ID	MPIDR format affinity value.

6.7.2 Base Platform DebugAccessPort component

This section describes the DebugAccessPort component which models the Debug Access Port (DAP) for external debug connections.

6.7.2.1 DebugAccessPort ports

This section describes the ports.

Table 6-19: DebugAccessPort ports

Name	Protocol	Type	Description
ap_pvbuss_m[2]	PVBus	Master	Debug-access port to bus master channels 0 and 1.
clock	ClockSignal	Slave	Clock input.
paddrdbg31	Signal	Master	Output signal that indicates which master the access came from, AP0 or AP1. Configurable.

6.7.2.2 DebugAccessPort parameters

This section describes the parameters.

Table 6-20: Base Platform DebugAccessPort parameters

Name	Type	Allowed values	Default value	Description
ap0_rom_base_address	uint64_t	0-0xffffffffffffffff	0	ROM base address for AP0.
ap1_rom_base_address	uint64_t	0-0xffffffffffffffff	0	ROM base address for AP1.
ap0_has_debug_rom	bool	true, false	false	AP0 has a Debug ROM.
ap1_has_debug_rom	bool	true, false	false	AP1 has a Debug ROM.

Name	Type	Allowed values	Default value	Description
ap0_set_paddrdbg31	bool	true, false	false	Set paddrdbg31 signal during accesses on AP0.
ap1_set_paddrdbg31	bool	true, false	false	Set paddrdbg31 signal during accesses on AP1.

6.7.3 Simulator visualization - parameters

This section describes the parameters.

Table 6-21: Simulator visualization parameters

Parameter	Allowed values	Default value	Description
cluster0_name	-	"Cluster0"	Cluster0 name.
cluster1_name	-	"Cluster1"	Cluster1 name.
cluster2_name	-	"Cluster2"	Cluster2 name.
cluster3_name	-	"Cluster3"	Cluster3 name.
cpu_name	-	""	Processor name displayed in window title.
diagnostics	true, false	false	Print diagnostic messages.
disable_visualisation	true, false	false	Enable or disable visualization.
rate_limit-enable	true, false	false	Restrict simulation speed so that simulation time more closely matches real time rather than running as fast as possible.
recorder. checkInstructionCount	true, false	true	Checks instruction count in recording file against actual instruction count during playback.
recorder. playbackFileName	-	""	Playback filename. An empty string disables playback.
recorder. recordingFileName	-	""	Recording filename. An empty string disables recording.
recorder. recordingTimeBase	-	0x5F5E100	Timebase in 1/s relative to the master clock. Used for recording. For example, 100 000 000 means 10 nanoseconds resolution simulated time for a 1Hz master clock. Higher values give a higher time resolution. Playback time base is always taken from the playback file.
recorder.verbose	-	0	Enables verbose messages. 0, no messages. 1, normal. 2, more detail.
trap_key	Valid ATKeyCode key value. See \$PVLIB_HOME/include/components/KeyCode.h for a list of values.	0x4a	Trap key that works with left Ctrl to toggle mouse display. The default is the left Alt key, so pressing Left Alt and Left Ctrl simultaneously toggles the mouse display.
window_title	-	"Fast Models - CLCD %cpu%"	Window title. cpu_name replaces %cpu%.

6.7.4 VE_SysRegs component

This section describes the VE system registers component.

6.7.4.1 VE_SysRegs parameters

This table describes the VE_SysRegs parameters.

Table 6-22: VE_SysRegs parameters

Name	Type	Default value	Description
exit_on_shutdown	bool	false	Used to shut down the system. See the Note after the table.
mmbSiteDefault	int	1	Default MultiMedia Bus (MMB) source (0=motherboard, 1=daughterboard 1, 2=daughterboard 2).
sys_proc_id0	int	0x0c000000	Processor ID register at CoreTile Express Site 1.
sys_proc_id1	int	0xff000000	Processor ID at CoreTile Express Site 2.
tilePresent	bool	true	Tile fitted.
user_switches_value	int	0	User switches.



When `exit_on_shutdown` is `true`, if software uses the `SYS_CFGCTRL` function `SYS_CFG_SHUTDOWN`, then the simulator shuts down and exits. For more information on the `SYS_CFGCTRL` function values, see the Motherboard Express µATX V2M-P1 Technical Reference Manual.

6.7.4.2 VE_SysRegs registers

This table describes the configuration registers.

Table 6-23: VE_SysRegs registers

Name	Offset	Access	Description
SYS_ID	0x00	Read/write	System identity
SYS_SW	0x04	Read/write	Bits[7:0] map to switch S6
SYS_LED	0x08	Read/write	Bits[7:0] map to user LEDs
SYS_100HZ	0x24	Read only	100Hz counter
SYS_FLAGS	0x30	Read/write	General purpose flags
SYS_FLAGSCLR	0x34	Write only	Clear bits in general purpose flags
SYS_NVFLAGS	0x38	Read/write	General purpose non-volatile flags
SYS_NVFLAGSCLR	0x3C	Write only	Clear bits in general purpose non-volatile flags
SYS_MCI	0x48	Read only	MCI
SYS_FLASH	0x4C	Read/write	Flash control
SYS_CFGSW	0x58	Read/write	Boot select switch
SYS_24MHZ	0x5C	Read only	24MHz counter

Name	Offset	Access	Description
SYS_MISC	0x60	Read/write	Miscellaneous control flags
SYS_DMA	0x64	Read/write	DMA peripheral map
SYS_PROCID0	0x84	Read/write	Processor ID
SYS_PROCID1	0x88	Read/write	Processor ID
SYS_CFGDATA	0xA0	Read/write	Data to be read/written from/to motherboard controller
SYS_CFGCTRL	0xA4	Read/write	Control data transfer to motherboard controller
SYS_CFGSTAT	0xA8	Read/write	Status of data transfer to motherboard

6.8 Base Platform VE compatibility

Arm expects software that ran on the previous VE model to be compatible with this system model, but you might need to apply some configuration options.

6.8.1 Base Platform GICv2

This system model uses GICv3 by default. You can configure it to support GICv2 or GICv2m.

To configure the model as GICv2m, set the following parameters:

```
-C gicv3.gicv2-only=1 \
-C cluster0.gic.GICD-offset=0x10000 \
-C cluster0.gic.GICC-offset=0x2F000 \
-C cluster0.gic.GICH-offset=0x4F000 \
-C cluster0.gic.GICH-other-CPU-offset=0x50000 \
-C cluster0.gic.GICV-offset=0x6F000 \
-C cluster0.gic.PERIPH-size=0x80000 \
-C cluster1.gic.GICD-offset=0x10000 \
-C cluster1.gic.GICC-offset=0x2F000 \
-C cluster1.gic.GICH-offset=0x4F000 \
-C cluster1.gic.GICH-other-CPU-offset=0x50000 \
-C cluster1.gic.GICV-offset=0x6F000 \
-C cluster1.gic.PERIPH-size=0x80000 \
-C gic_distributor.GICD-alias=0x2c010000
```

To configure the model as GICv2, set the following parameters:

```
-C gicv3.gicv2-only=1 \
-C cluster0.gic.GICD-offset=0x1000 \
-C cluster0.gic.GICC-offset=0x2000 \
-C cluster0.gic.GICH-offset=0x4000 \
-C cluster0.gic.GICH-other-CPU-offset=0x5000 \
-C cluster0.gic.GICV-offset=0x6000 \
-C cluster0.gic.PERIPH-size=0x8000 \
-C cluster1.gic.GICD-offset=0x1000 \
-C cluster1.gic.GICC-offset=0x2000 \
-C cluster1.gic.GICH-offset=0x4000 \
-C cluster1.gic.GICH-other-CPU-offset=0x5000 \
-C cluster1.gic.GICV-offset=0x6000 \
-C cluster1.gic.PERIPH-size=0x8000 \
-C gic_distributor.GICD-alias=0x2c010000
```

To configure MSI frames for GICv2m, parameters are available to set the base address and configuration of each of 16 possible frames. Eight frames are Secure and eight frames are Non-secure:

```
-C gic_distributor.MSI_S-frame0-base=ADDRESS \
-C gic_distributor.MSI_S-frame0-min-SPI=NUM \
-C gic_distributor.MSI_S-frame0-max-SPI=NUM
```

In this example, you can replace `msi_s` with `MSI_NS`, for NS frames, and you can replace `frame0` with `frame1` to `frame7` for each of the possible 16 frames. If the base address is not specified for a given frame, or the SPI numbers are out of range, the corresponding frame is not instantiated.

6.8.2 Base Platform GICv3

If a Base Platform includes an implementation of the GICv3 system registers, it is enabled by default.

The GIC distributor and CPU (core) interface have parameters that allow configuration of the model to match different implementation options. Use `--list-params` to get a full list. Configuration options for the GIC model must be available under:

- `cluster[0-n].gic.*`
- `cluster[0-n].gicv3.*`
- `gic_distributor.*`

6.8.3 Base Platform system global counter

The Generic Timer registers of the cores do not operate by default.

The model provides a memory-mapped interface to the system global counter, and enables the free-running timer from reset. However, the architectural requirement is that such a counter is not enabled at reset. As a result, the Generic Timer registers of the cores do not operate unless either:

- Software enables the counter peripheral by writing the `FCREQ[0]` and `EN` bits in `CNTCR` at `0x2a430000`. Arm recommends this approach.
- The `-c bp.refcounter.non_arch_start_at_default=1` parameter is set. This approach provides compatibility with older software.

6.8.4 Disable security in the Base Platform

Base Platform FVPs have an enhanced security map for peripherals. By default, it restricts access to some peripherals.

Software must program the TZC-400 to make any accesses to DRAM, because the reset configuration blocks all accesses.

For backward compatibility with software that cannot program the TZC-400, this parameter setting permits all accesses regardless of security state:

```
-C bp.secure_memory=false
```

Related information

[Base Platform memory map](#) on page 4230

6.9 Base Platform unsupported VE features

This system model does not support software that relies on some features of the VE model.

6.9.1 Base Platform memory aliasing at 0x08_00000000

The VE model permits an alias of the 2GB region of DRAM between addresses 0x80000000 and 0xFFFFFFFF with addresses 0x08_00000000 to 0x08_7FFFFFFF. The Base Platform does not have this alias and the region 0x08_00000000 to 0x08_7FFFFFFF is Reserved.

6.9.2 Boot ROM alias at 0x00_0800_0000 in the Base Platform

In the VE model, the region at 0x00_0800_0000 was an alias of the trusted boot ROM at 0x00_0000_0000. It is now an independent region of NOR flash.

6.9.3 Changed parameters in the Base Platform

Most parameter names in the VE model have been simplified in the Base Platform.

Components that were previously in motherboard or daughterboard groups are now in a `bp` group. The model does not recognize the previous parameter names.

In a change to the previous default, the Base Platform models the core cache state by default. You can disable this using a single parameter for all cores in the simulation, using the `cache_state_modelled` parameter:

```
-C cache_state_modelled=0
```



Cortex® Base Platforms do not model the cache state by default.

7. Microcontroller Prototyping System 2

This chapter describes the MPS2 system model.

7.1 MPS2

The Microcontroller Prototyping System 2 (MPS2) Fixed Virtual Platforms (FVPs) implement a subset of the functionality of the Arm Versatile Express V2M-MPS2 and V2M-MPS2+ motherboard hardware.

For a list of the MPS2 Platform FVPs and the instances in them, see [MPS2 Platform FVPs](#) in the FVP Reference Guide.

LISA+ source and project files for some MPS2 FVP examples are available in the `$FVLIB_HOME/examples/LISA/FVP_MPS2/` directory.

MPS2 platforms include MPS2-specific components and generic components, such as buses and timers. They are sufficiently accurate to boot the Keil RTX RTOS and run the Blinky application.

To list the model parameters and their types, allowed values, default values, and descriptions, run the model with the `--list-params` argument.

Related information

[AN400 - ARM Cortex-M7 SMM on V2M-MPS2](#)

7.2 MPS2 platform types

Configure the MPS2 FVP platform type using the `fvf_mps2.platform_type` parameter.

It has the following possible values:

0

This value is the default. The FVP acts as a V2M-MPS2 system, with the additions for v8-M, as specified in the Arm®v8-M MPS2 System Specification (ECM 0468897), v0.8. This specification is confidential and is available only to licensed Arm customers. For details, contact your Arm support representative.

1

The FVP acts as an IoT Kit on an MPS2+ board. For details, see the following documents:

- [Cortex®-M23 processor Arm®v8-M IoT Kit User Guide \(ECM 0635473\)](#).
- [Cortex®-M33 processor Arm®v8-M IoT Kit User Guide \(ECM 0601256\)](#).

2

The FVP acts as an Arm® CoreLink™ SSE-200 Subsystem on an MPS2+ board. For details, see [AN521 - Example SSE-200 Subsystem for MPS2+ Application Note](#).

7.3 MPS2 memory maps

This section describes the MPS2 memory maps.

7.3.1 MPS2 memory map for models without the Armv8-M additions

This section describes the MPS2 memory map for older cores, without the Armv8-M additions.

For standard Arm peripherals, see the TRM for that device.



- A bus error is generated for accesses to memory areas not shown in this table.
- Any memory device that does not occupy the total region is aliased within that region.

Table 7-1: Memory map for models without the Armv8-M additions

Description	Modeled	Address range
Ethernet. Through ahb_to_extmem16. Offset 0x0 to 0x0FE for CSRs, 0x100 to 0x1FE for FIFO.	Partial	0xA0000000 to 0xA00FFFFF
PSRAM (16MB)	Yes	0x60000000 to 0x60FFFFFF
VGA Image (512x128) (AHB)	Yes	0x41100000 to 0x4110FFFF
VGA Console (AHB)	Yes	0x41000000 to 0x4100FFFF
Block RAM (boot time). Reserved 64KB, 16K implemented. This memory is wrapped through the region.	Yes	0x40200000 to 0x402FFFFF
Reserved	N/A	0x40030000 to 0x401FFFFF
SCC register	Yes	0x4002F000 to 0x4002FFFF
Reserved	N/A	0x40029000 to 0x4002EFFF
FPGA System Control & I/O, APB	Yes	0x40028000 to 0x40028FFF
Reserved	N/A	0x40025000 to 0x40027FFF
Audio I2S, APB	Partial	0x40024000 to 0x40024FFF
SBCon (Audio Configuration), APB	Yes	0x40023000 to 0x40023FFF
SBCon (Touch for LCD module), APB	Partial	0x40022000 to 0x40022FFF
PL022 (SPI for LCD module), APB	Partial	0x40021000 to 0x40021FFF

Description	Modeled	Address range
PL022 (SPI), APB	Yes	0x40020000 to 0x40020FFF
CMSDK system controller	Yes	0x4001F000 to 0x4001FFFF
Reserved for extra GPIO & other AHB peripherals	N/A	0x40012000 to 0x4001EFFF
CMSDK AHB GPIO #1	Yes	0x40011000 to 0x40011FFF
CMSDK AHB GPIO #0	Yes	0x40010000 to 0x40010FFF
CMSDK APB subsystem	Yes	0x40000000 to 0x4000FFFF
Reserved	N/A	0x20800000 to 0x20FFFFFF
ZBTSRAM 2 & 3 (2x32-bit). Reserved 8MB, 4MB available. The two SRAM blocks are interleaved.	Yes	0x20000000 to 0x207FFFFF
Reserved	N/A	0x01010000 to 0x1FFFFFFF
Reserved	N/A	0x00800000 to 0x00FFFFFF
ZBTSRAM 1 (64-bit). Wrapped. Only 4MB ZBTSRAM fitted.	Yes	0x00400000 to 0x007FFFFF
ZBTSRAM 1 (64-bit)	Yes	0x00004000 to 0x003FFFFF
Mappable memory. When <code>zbt_boot_ctrl = 0</code> , ZBTSRAM 1 is mapped to this region. Otherwise, <code>Remap_ctrl = 0</code> maps Block RAM and <code>Remap_ctrl = 1</code> maps ZBTSRAM 1. The V2M-MPS2 board microcontroller controls the <code>zbt_boot_ctrl</code> signal. The <code>zbt_boot_ctrl</code> signal overrides the boot option to enable use of the ZBT RAM.	Yes	0x00000000 to 0x00003FFF

7.3.2 MPS2 memory map for models with the Armv8-M additions

This section describes the MPS2 memory map for newer cores, with the Armv8-M additions.

For standard Arm peripherals, see the TRM for that device.



- A bus error is generated for accesses to memory areas not shown in this table.
- Any memory device that does not occupy the total region is aliased within that region.

Table 7-2: Memory map for models with the Armv8-M additions

Description	IDAU	Modeled	Address range
ZBTSRAM 1 (4MB) in Non-secure (NS) world. Reserved 8MB, only 4MB implemented. VTOR initialization value to be configurable in LAC. Second half (4MB) aliased to first half (4MB).	NS	Yes	0x00000000 to 0x007FFFFFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0x00800000 to 0x0FFFFFFF
ZBTSRAM 1 (4MB) in Secure (S) world. Reserved 8MB, only 4MB implemented. Second half (4MB) aliased to first half (4MB).	S	Yes	0x10000000 to 0x107FFFFFFF
Not used. Default expansion port: bus error.	S	N/A	0x10800000 to 0x1FFFFFFF
ZBTSRAM 2 and ZBTSRAM 3 (4MB) in NS world. Reserved 8MB, only 4MB implemented. For IoT subsystems, different cores have different memory sizes. Second half (4MB) aliased to first half (4MB).	NS	Yes	0x20000000 to 0x207FFFFFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0x20800000 to 0x20FFFFFFF
PSRAM (32MB)	NS	Yes	0x21000000 to 0x22FFFFFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0x23000000 to 0x23FFFFFFF
MTB SRAM. Reserved 64KB, only 16KB implemented. Aliased to 0x0 for booting in RTL simulation.	NS	Yes	0x24000000 to 0x2400FFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0x24010000 to 0x2FFFFFFF
ZBTSRAM 2 and ZBTSRAM 3 (4MB) in S world. Reserved 8MB, only 4MB implemented. Second half (4MB) aliased to first half (4MB).	S	Yes	0x30000000 to 0x307FFFFFFF
Not used. Default expansion port: bus error.	S	N/A	0x30800000 to 0x30FFFFFFF
Not used. No memory gating unit on PSRAM (16MB) path because it is shared with Ethernet control. Default expansion port: bus error.	S	N/A	0x31000000 to 0x31FFFFFFF
Not used. Default expansion port: bus error.	S	N/A	0x30800000 to 0x3FFFFFFF
Timer 0. Non-secure CMSDK APB subsystem.	NS	Yes	0x40000000 to 0x40000FFF
Timer 1. Non-secure CMSDK APB subsystem.	NS	Yes	0x40001000 to 0x40001FFF
Dual Timer. Non-secure CMSDK APB subsystem.	NS	Yes	0x40002000 to 0x40002FFF
Not used. Non-secure CMSDK APB subsystem.	NS	N/A	0x40003000 to 0x40003FFF
UART #0. Non-secure CMSDK APB subsystem.	NS	Yes	0x40004000 to 0x40004FFF
UART #1. Non-secure CMSDK APB subsystem.	NS	Yes	0x40005000 to 0x40005FFF
UART #2. Non-secure CMSDK APB subsystem.	NS	Yes	0x40006000 to 0x40006FFF

Description	IDAU	Modeled	Address range
Not used. Non-secure CMSDK APB subsystem.	NS	N/A	0x40007000 to 0x40007FFF
Watchdog. Non-secure CMSDK APB subsystem.	NS	Yes	0x40008000 to 0x40008FFF
Not used. Non-secure CMSDK APB subsystem.	NS	N/A	0x40009000 to 0x4000F000
GPIO #0.	NS	Yes	0x40010000 to 0x40010FFF
GPIO #1.	NS	Yes	0x40011000 to 0x40011FFF
GPIO #2.	NS	Yes	0x40012000 to 0x40012FFF
GPIO #3.	NS	Yes	0x40013000 to 0x40013FFF
Default slave inside AHB peripheral subsystem. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	Yes	0x40014000 to 0x40017FFF
DMA Controller #0.	NS	Yes	0x40018000 to 0x40018FFF
DMA Controller #1.	NS	Yes	0x40019000 to 0x40019FFF
Default slave inside AHB peripheral subsystem. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	Yes	0x4001A000 to 0x4001EFFF
CMSDK system controller. PMU control and remap registers unused. Only reset option (lockup reset) and rest info available.	NS	Yes	0x4001F000 to 0x4001FFFF
Not used. Non-secure FPGA APB subsystem (unused APB space: RAZ/WI).	NS	N/A	0x40020000 to 0x40020FFF
PL022 (SPI for LCD). Non-secure FPGA APB subsystem (unused APB space: RAZ/WI).	NS	Partial	0x40021000 to 0x40021FFF
SBCon I2C (Touch for LCD). Non-secure FPGA APB subsystem (unused APB space: RAZ/WI).	NS	Partial	0x40022000 to 0x40022FFF
SBCon I2C (Audio configuration). Non-secure FPGA APB subsystem (unused APB space: RAZ/WI).	NS	Yes	0x40023000 to 0x40023FFF
Audio I2S. Non-secure FPGA APB subsystem (unused APB space: RAZ/WI).	NS	Partial	0x40024000 to 0x40024FFF
Not used. Non-secure FPGA APB subsystem (unused APB space: RAZ/WI).	NS	N/A	0x40025000 to 0x40027FFF
FPGA system control & I/O (LEDs, buttons...). Non-secure FPGA APB subsystem (unused APB space: RAZ/WI).	NS	Yes	0x40028000 to 0x40028FFF
Not used. Non-secure FPGA APB subsystem (unused APB space: RAZ/WI).	NS	N/A	0x40029000 to 0x4002EFFF
SCC registers. Non-secure FPGA APB subsystem (unused APB space: RAZ/WI).	NS	Yes	0x4002F000 to 0x4002FFFF
Not used.	NS	N/A	0x40030000 to 0x40113FFF
SVOS DualTimer. Only enabled for Cortex-M55 SVOS.	NS	Yes	0x40114000 to 0x40114fff

Description	IDAU	Modeled	Address range
Not used.	NS	N/A	0x40115000 to 0x401FFFFFF
Ethernet (SMSC 91C111).	NS	Partial	0x40200000 to 0x402FFFFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0x40300000 to 0x40FFFFFFF
VGA console.	NS	Yes	0x41000000 to 0x4100FFFF
Not used.	NS	N/A	0x41010000 to 0x410FFFFFF
VGA image.	NS	Yes	0x41100000 to 0x4113FFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0x41140000 to 0x4FFFFFFF
Timer 0. Secure CMSDK APB subsystem.	S	Yes	0x50000000 to 0x50000FFF
Timer 1. Secure CMSDK APB subsystem.	S	Yes	0x50001000 to 0x50001FFF
Dual Timer. Secure CMSDK APB subsystem.	S	Yes	0x50002000 to 0x50002FFF
Not used. Secure CMSDK APB subsystem.	S	N/A	0x50003000 to 0x50003FFF
UART #0. Secure CMSDK APB subsystem.	S	Yes	0x50004000 to 0x50004FFF
UART #1. Secure CMSDK APB subsystem.	S	Yes	0x50005000 to 0x50005FFF
UART #2. Secure CMSDK APB subsystem.	S	Yes	0x50006000 to 0x50006FFF
Not used. Secure CMSDK APB subsystem.	S	N/A	0x50007000 to 0x50007FFF
Watchdog. Secure CMSDK APB subsystem.	S	Yes	0x50008000 to 0x50008FFF
Not used. Secure CMSDK APB subsystem.	S	N/A	0x50009000 to 0x5000F000
GPIO #0	S	Yes	0x50010000 to 0x50010FFF
GPIO #1	S	Yes	0x50011000 to 0x50011FFF
GPIO #2	S	Yes	0x50012000 to 0x50012FFF
GPIO #3	S	Yes	0x50013000 to 0x50013FFF
Default slave. Default expansion port: bus error.	S	Yes	0x50014000 to 0x50017FFF
DMA Controller #0.	S	Yes	0x50018000 to 0x50018FFF

Description	IDAU	Modeled	Address range
DMA Controller #1.	S	Yes	0x50019000 to 0x50019FFF
Default slave. Default expansion port: bus error.	S	Yes	0x5001A000 to 0x5001EFFF
CMSDK system controller. PMU control and remap registers unused. Only reset option (lockup reset) and rest info available.	S	Yes	0x5001F000 to 0x5001FFFF
FPGA APB subsystem	S	-	0x50020000 to 0x5002FFFF
Not used. Secure FPGA APB subsystem	S	N/A	0x50020000 to 0x50020FFF
PL022 (SPI for LCD). Secure FPGA APB subsystem	S	Partial	0x50021000 to 0x50021FFF
SBCon I2C (touch for LCD). Secure FPGA APB subsystem	S	Partial	0x50022000 to 0x50022FFF
SBCon I2C (audio configuration). Secure FPGA APB subsystem	S	Yes	0x50023000 to 0x50023FFF
Audio I2S. Secure FPGA APB subsystem	S	Partial	0x50024000 to 0x50024FFF
Not used. Secure FPGA APB subsystem	S	N/A	0x50025000 to 0x50027FFF
FPGA system control & I/O (LEDs, buttons...). Secure FPGA APB subsystem	S	Yes	0x50028000 to 0x50028FFF
Not used. Secure FPGA APB subsystem	S	N/A	0x50029000 to 0x5002EFFF
SCC registers. Secure FPGA APB subsystem	S	Yes	0x5002F000 to 0x5002FFFF
Not used.	S	N/A	0x50030000 to 0x50113FFF
SVOS DualTimer. Only enabled for Cortex-M55 SVOS.	S	Yes	0x50114000 to 0x50114fff
Not used.	S	N/A	0x50115000 to 0x501FFFFF
Ethernet (SMSC 91C111).	S	Partial	0x50200000 to 0x502FFFFF
Not used. Default expansion port: bus error.	S	N/A	0x50300000 to 0x50FFFFFF
VGA console.	S	Yes	0x51000000 to 0x5100FFFF
Not used.	S	N/A	0x51010000 to 0x510FFFFFFF
VGA image.	S	Yes	0x51100000 to 0x5113FFFF
Not used.	S	N/A	0x51140000 to 0x58006FFF
Secure Control Registers. Secure APB subsystem.	S	Yes	0x58007000 to 0x58007FFF

Description	IDAU	Modeled	Address range
Flash memory gating unit configuration (mapped to AHB port for CODE region in the bus matrix, not APB). Secure APB subsystem.	S	Yes	0x58008000 to 0x58009FFF
SRAM memory gating unit configuration (mapped to AHB port for SRAM region in the bus matrix, not APB). Secure APB subsystem.	S	Yes	0x5800A000 to 0x5800DFFF
Reserved. Secure APB subsystem.	S	N/A	0x5800E000 to 0x5800EFFF
Reserved. Secure APB subsystem.	S	N/A	0x5800F000 to 0x5800FFFF
Not used. Default expansion port: bus error.	S	N/A	0x58010000 to 0x5FFFFFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0x60000000 to 0x6FFFFFFF
Not used. Default expansion port: bus error.	S	N/A	0x70000000 to 0x7FFFFFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0x80000000 to 0x8FFFFFFF
Not used. Default expansion port: bus error.	S	N/A	0x90000000 to 0x9FFFFFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0xA0000000 to 0xAFFFFFFF
Not used. Default expansion port: bus error.	S	N/A	0xB0000000 to 0xBFFFFFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0xC0000000 to 0xCFFFFFFF
Not used. Default expansion port: bus error.	S	N/A	0xD0000000 to 0xDFFFFFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0xE0000000 to 0xEFFFFFFF
System ROM table. Exempted from checking.	Exempt	Yes	0xF0000000 to 0xF0000FFF
Not used. Default expansion port: bus error.	Exempt	N/A	0xF0001000 to 0xF00FFFFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0xF0100000 to 0xF01FFFFF
MTB SFR address space.	NS	Yes	0xF0200000 to 0xF0200FFF
Reserved. This region is non-executable.	NS	N/A	0xF0210000 to 0xF0213FFF
Not used. Default expansion port (MPS2 AHB subsystem): bus error or RAZ/WI .	NS	N/A	0xF0214000 to 0xFFFFFFFF

7.4 MPS2 interrupt assignments

This section describes the interrupt assignments.

Table 7-3: Interrupt assignments

Number	Interrupt
NMI	Watchdog.
0	UART 0 receive interrupt.
1	UART 0 transmit interrupt.
2	UART 1 receive interrupt.
3	UART 1 transmit interrupt.
4	UART 2 receive interrupt.
5	UART 2 transmit interrupt.
6	GPIO 0, 2 combined interrupt.
7	GPIO 1, 3 combined interrupt.
8	Timer 0.
9	Timer 1.
10	Dual Timer.
11	SPI #1 (LCD). The LCD had shared SPI #0 and SPI #1.
12	UART overflow (0, 1, 2).
13	Ethernet.
14	Audio I2S.
15	Touch screen.
16-31	GPIO 0 individual interrupts.
32-47	GPIO 1 individual interrupts. ArmR8-M additions.
48	SPI #0. ArmR8-M addition.
49	Reserved.
50	TRNG (Secure). ArmR8-M addition.
51	Unique ID and Secure storage (Secure). ArmR8-M addition.
52	DMA controller #0.
53	DMA controller #1.
54	SecureErrorIRQ. ArmR8-M addition. Detection of Non-secure access to Secure address spaces (including other bus masters). Generated by Memory Gating unit, Peripheral Gating units, bus gasket for legacy bus masters.

7.5 MPS2 - differences between models and hardware

This section describes the features of the hardware that the models do not implement, or implement with significant differences.

MPS2 implements most devices. Some peripherals have minimal implementations:

- The Ethernet module in the model is a LAN91C111. The hardware provides a LAN9220.

- The Audio module is **RAZ/WI**.
- The STMPE811 touchscreen module only reports touch positions.
- The model of the Ampire LCD module supports a subset of the graphics modes.

You can display images and text on an emulated VGA output, images on the LCD, and text on the UART.

RX overrun mode

The CMSDK_UART component has a parameter `rx_overrun_mode` that controls how to handle the transfer of characters into the UART RX FIFO when the `RX_OVERRUN` flag is set in the STATE register. It has the following possible values:

0

Never block the transfer of the next character into the RX FIFO even if it means losing multiple characters. This is the same behavior as the hardware. It might be useful when evaluating software design, for example to indicate whether the UART can be serviced quickly enough.

In this mode, as in the hardware, the `RX_OVERRUN` flag serves only to alert software to the fact that characters have been lost and that some sort of corrective action, or recovery procedure might be required.

1

Pause the transfer of characters into the RX FIFO before any characters are lost. This makes the serial connection lossless even if the software does not make any attempt to service the UART in a timely fashion.

In this mode, the transfer of characters can be resumed by reading a character from the DATA register, so no special action is required by software.

2

The default value. Pause the transfer of characters into the UART RX FIFO after the first character is lost due to overrun.

In this mode, clearing the `RX_OVERRUN` flag resumes the transfer of characters. This requires the software to write to the STATE or INT register to clear the flag. Precisely one character will have been lost. This is the legacy behavior of the UART.

Arm®v8-M

The model does not support MTB, ETM, and TPIU. MTB RAM is absent.

In the Memory Gating Unit, the model provides a configurable block size. For performance reasons, the minimum block size in the model is 4096 bytes. Hardware and later models might allow smaller block sizes. Software must use the `BLK_CFG` register to determine block size.

Timing

FVPs enable software applications to run in a functionally accurate simulation. However, because of the relative balance of fast simulation speed over timing accuracy, there are situations where the models might behave unexpectedly.

If your code interacts with real world devices such as timers and keyboards, data arrives in the modeled device in real world, or wall clock, time. However, simulation time can run faster than the wall clock. So, a single key press might be interpreted as several repeated key presses, or a single mouse click might be interpreted as a double click.

To avoid this mismatch, the FVPs provide the Rate Limit feature. Enabling Rate Limit forces the model to run at wall clock time. For interactive applications, Arm recommends enabling Rate Limit. Use the Rate Limit button in the CLCD display or the `rate_limit-enable` model instantiation parameter.

8. Versatile Express Model

This chapter describes the components of Fast Models that are specific to the Versatile™ Express (VE) model of the hardware platform.

8.1 VE

The Versatile Express (VE) FVPs are functionally-accurate system models for software execution. A range of VE FVPs are supplied as standalone products and as examples in Fast Models.

For a list of the VE Platform FVPs that are included in the FVP Standard Library package, see [VE Platform FVPs](#) in the FVP Reference Guide. The package can be downloaded from [Fixed Virtual Platforms](#).

LISA+ source and project files for the VE FVPs are available in the `$PVLIB_HOME/examples/LISA/FVP_VE/` directory.

Arm produces the VE hardware development platform. The Motherboard Express µAdvanced Technology Extended (ATX) V2M-P1 is the basis for an integrated software and hardware development system. This system is also based on the Arm Symmetric MultiProcessor (SMP) system architecture.

The VE FVPs are system models implemented in software. Each model contains:

- A virtual implementation of the Arm VE motherboard.
- A single daughterboard containing one or more Arm processors.
- Associated interconnections.

The motherboard provides:

- Peripherals for multimedia or networking environments.
- Access to motherboard peripherals and functions through a static memory bus to simplify access from daughterboards.
- High-performance PCI-Express slots for expansion cards.
- Consistent memory maps with different processor daughterboards that simplify software development and porting.
- Automatic detection and configuration of attached CoreTile Express and LogicTile Express daughterboards.
- Automatic shutdown for over-temperature or power supply failure.
- No system power-on for unconfigurable daughterboards.
- Power sequencing of system.
- Drag and drop file updating of configuration files.
- Support of either a 12V power-supply unit or an external ATX power supply.

- Support of FPGA and processor daughterboards to provide custom peripherals, early access to processor designs, or production test chips.

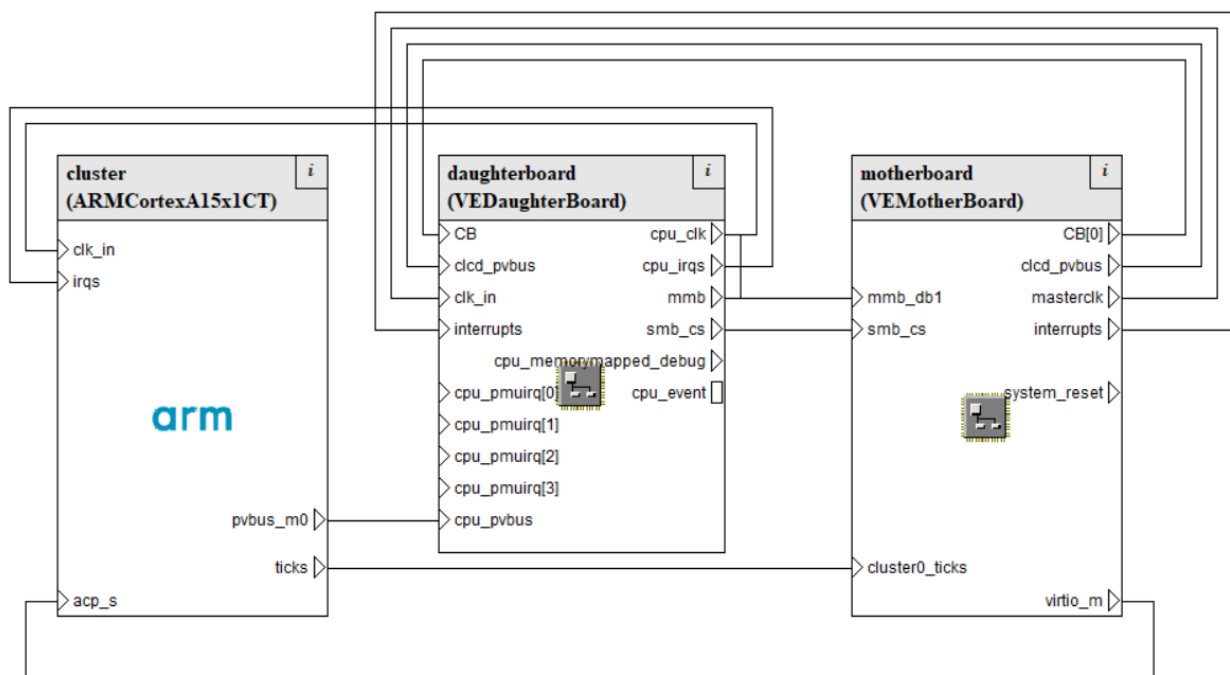


Arm bases the models on the VE platform memory map, but does not intend them to be accurate representations of a specific VE hardware revision. The VE FVPs support selected peripherals. The models are sufficiently complete and accurate to boot the same operating system images as the VE hardware.

VE FVPs provide functionally accurate models for software execution. However, the models sacrifice timing accuracy to increase simulation speed. Key deviations from hardware are:

- Approximate timing.
- Simplified buses.
- No implementations for processor caches and the related write buffers.

Figure 8-1: Top level block diagram of a VE model



Related information

- [Getting Started with Fixed Virtual Platforms](#)
- [ARM Motherboard Express µATX V2M-P1 Technical Reference Manual](#)

8.2 VE memory map for Cortex-A series

The global memory map for the Cortex®-A series VE model is based on the hardware Versatile™ Express RS1 memory map with the RS2 extensions.



Note

The VE FVP implementation of memory does not require the memory controller to have the correct values. If you run applications on hardware, ensure that the memory controller is set up properly. Otherwise, applications that run on the FVP might fail on hardware.

Table 8-1: Cortex®-A series platform model memory map

Name	Modeled	Address range	Size
NOR FLASH0 (CS0)	Yes	0x00_00000000-0x00_03FFFFFF	64MB
Reserved	-	0x00_04000000-0x00_07FFFFFF	64MB
NOR FLASH0 alias (CS0)	Yes	0x00_08000000-0x00_0BFFFFFF	64MB
NOR FLASH1 (CS4)	Yes	0x00_0C000000-0x00_0FFFFFFF	64MB
Unused (CS5)	-	0x00_10000000-0x00_13FFFFFF	-
PSRAM (CS1) - unused	No	0x00_14000000-0x00_17FFFFFF	-
Peripherals (CS2). See Table: CS2 region peripheral memory map, below.	Yes	0x00_18000000-0x00_1BFFFFFF	64MB
Peripherals (CS3). See Table: CS2 region peripheral memory map, below.	Yes	0x00_1C000000-0x00_1FFFFFFF	64MB
CoreSight™ and peripherals	No	0x00_20000000-0x00_2CFFFFFF. See Note after table.	-
Graphics space	No	0x00_2D000000-0x00_2D00FFFF	-
System SRAM	Yes	0x00_2E000000-0x00_2EFFFFFF	64KB
Ext AXI	No	0x00_2F000000-0x00_7FFFFFFF	-
4GB DRAM (in 32-bit address space). See Note after table.	Yes	0x00_80000000-0x00_FFFFFFFF	2GB
Unused	-	0x01_00000000-0x07_FFFFFFFF	-
4GB DRAM (in 36-bit address space). See Note after table.	Yes	0x08_00000000-0x08_FFFFFFFF	4GB
Unused	-	0x09_00000000-0x7F_FFFFFFFF	-
4GB DRAM (in 40-bit address space). See Note after table.	Yes	0x80_00000000-0xFF_FFFFFFFF	4GB



Note

- The private peripheral address 0x2c000000 is mapped in the CoreSight™ and peripherals region. Use the parameter PERIPHBASE to map the peripherals to a different address.
- The model contains a single 4GB block of DRAM, which is aliased across the three different DRAM regions. In other words, it can be accessed at three different physical addresses, which are all mapped to the same area of DRAM. For example, a write to address 0x00_80000000 is visible to reads at address 0x80_00000000. The lowest of the physical address regions is only 2GB in size.

The model has a `secure_memory` option. When you enable this option, the memory map changes for a number of peripherals.

Table 8-2: CS2 region peripheral memory map for `secure_memory` option

Peripheral	Address range	Functionality with <code>secure_memory</code> enabled
NOR FLASH0 (CS0)	0x00_00000000-0x00_0001FFFF	Secure RO, aborts on non-secure accesses.
Reserved	0x00_04000000-0x00_0401FFFF	Secure SRAM, aborts on non-secure accesses.
NOR FLASH0 alias (CS0)	0x00_08000000-0x00_7DFFFFFF	Normal memory map, aborts on secure accesses.
Ext AXI	0x00_7e000000-0x00_7FFFFFFF	Secure DRAM, aborts on non-secure accesses.
4GB DRAM (in 32-bit address space)	0x00_80000000-0xFF_FFFFFFFF	Normal memory map, aborts on secure accesses.

Table 8-3: CS2 region peripheral memory map

Peripheral	Modeled	Address range	Size	GIC Int. See Note after table.
VRAM - aliased	Yes	0x00_18000000-0x00_19FFFFFF	32MB	-
Ethernet (SMSC 91C111)	Yes	0x00_1A000000-0x00_1AFFFFFF	16MB	47
USB - unused	No	0x00_1B000000-0x00_1BFFFFFF	16MB	-



Use these interrupt signal values to program your interrupt controller. They are the SPI number plus 32. Add 32 to the interrupt number from the peripherals to form the interrupt number that the GIC sees. GIC interrupts 0-31 are for internal use.

Table 8-4: CS3 region peripheral memory map

Peripheral	Modeled	Address range	Size	GIC Int. See Note after table.
Local DAP ROM	No	0x00_1C000000-0x00_1C00FFFF	64KB	-
VE System Registers	Yes	0x00_1C010000-0x00_1C01FFFF	64KB	-
System Controller (SP810)	Yes	0x00_1C020000-0x00_1C02FFFF	64KB	-
TwoWire serial interface (PCIe)	No	0x00_1C030000-0x00_1C03FFFF	64KB	-
AACI (PL041)	Yes	0x00_1C040000-0x00_1C04FFFF	64KB	43
MCI (PL180)	Yes	0x00_1C050000-0x00_1C05FFFF	64KB	41, 42
KMI - keyboard (PL050)	Yes	0x00_1C060000-0x00_1C06FFFF	64KB	44
KMI - mouse (PL050)	Yes	0x00_1C070000-0x00_1C07FFFF	64KB	45
Reserved	-	0x00_1C080000-0x00_1C08FFFF	64KB	-
UART0 (PL011)	Yes	0x00_1C090000-0x00_1C09FFFF	64KB	37
UART1 (PL011)	Yes	0x00_1C0A0000-0x00_1C0AFFFF	64KB	38
UART2 (PL011)	Yes	0x00_1C0B0000-0x00_1C0BFFFF	64KB	39
UART3 (PL011)	Yes	0x00_1C0C0000-0x00_1C0CFFFF	64KB	40
Reserved	-	0x00_1C0D0000-0x00_1C0EFFFF	128KB	-
Watchdog (SP805)	Yes	0x00_1C0F0000-0x00_1C0FFFFFFF	64KB	32
Reserved	-	0x00_1C100000-0x00_1C10FFFF	64KB	-
Timer-0 (SP804)	Yes	0x00_1C110000-0x00_1C11FFFF	64KB	34
Timer-1 (SP804)	Yes	0x00_1C120000-0x00_1C12FFFF	64KB	35

Peripheral	Modeled	Address range	Size	GIC Int. See Note after table.
Virtio block device	Yes	0x00_1C130000-0x00_1C13FFFF	64KB	74
Virtio P9 device	Yes	0x00_1C140000-0x00_1C14FFFF	64KB	75
Reserved	-	0x00_1C130000-0x00_1C15FFFF	192KB	-
TwoWire serial interface (DVI) - unused	No	0x00_1C160000-0x00_1C16FFFF	64KB	-
Real-time Clock (PL031)	Yes	0x00_1C170000-0x00_1C17FFFF	64KB	36
Reserved	-	0x00_1C180000-0x00_1C19FFFF	128KB	-
CF Card - unused	No	0x00_1C1A0000-0x00_1C1AFFFF	64KB	-
Reserved	-	0x00_1C1B0000-0x00_1C1EFFFF	256KB	-
Color LCD Controller (PL111)	Yes	0x00_1C1F0000-0x00_1C1FFFFF	64KB	46
Reserved	-	0x00_1C200000-0x00_1FFFFFFF	64KB	-



Use these interrupt signal values to program your interrupt controller. They are the SPI number plus 32. Add 32 to the interrupt number from the peripherals to form the interrupt number that the GIC sees. GIC interrupts 0-31 are for internal use.

8.3 VE memory map for Cortex-R series

The Versatile™ Express RS1 memory map with the RS2 extensions is the base of the global memory map for the Cortex®-R series platform model.

Table 8-5: Cortex®-R series VE FVP memory map

Memory	Modeled	Address range
DRAM	Yes	0x00000000-0x3FFFFFFF
FLASH0	Yes	0x40000000-0x43FFFFFF
FLASH1	Yes	0x44000000-0x47FFFFFF
PSRAM	Yes	0x48000000-0x4BFFFFFF
RAM	No	0x4C000000-0x4FFFFFFF
PL390 GIC CPU Interface. Cortex®-R4 and Cortex®-R5 models only.	Yes	0xAE000000-0xAE000FFF
PL390 GIC Distributor. Cortex®-R4 and Cortex®-R5 models only.	Yes	0xAE001000-0xAE001FFF
VE System Registers	Yes	0xB0000000-0xB000FFFF
SP810	Yes	0xB0010000-0xB001FFFF
PL041 AACI	Yes	0xB0040000-0xB004FFFF
PL180 MCI	Yes	0xB0050000-0xB005FFFF
PL050 KMIO	Yes	0xB0060000-0xB006FFFF
PL050 KMI1	Yes	0xB0070000-0xB007FFFF
PL011 UART0	Yes	0xB0090000-0xB009FFFF
PL011 UART1	Yes	0xB00A0000-0xB00AFFFF
PL011 UART2	Yes	0xB00B0000-0xB00BFFFF
PL011 UART3	Yes	0xB00C0000-0xB00CFFFF

Memory	Modeled	Address range
SP805 WATCHDOG	Yes	0xB00F0000-0xB00FFFFF
TIMER_0_1	Yes	0xB0110000-0xB011FFFF
TIMER_2_3	Yes	0xB0120000-0xB012FFFF
PL031 Real Time Clock	Yes	0xB0170000-0xB017FFFF
Compact Flash	No	0xB01A0000-0xB01AFFFF
PL011 UART4	Yes	0xB01B0000-0xB01BFFFF
PL111 CLCD	Yes	0xB01F0000-0xB01FFFFF. See Note after table
RAM	No	0xB4000000-0xBBFFFFFFF
Video RAM	Yes	0xBC000000-0xBDFFFFFFF
Ethernet (SMSC 91C111)	Yes	0xBE000000-0xBEFFFFFFF
USB	No	0xBF000000-0xBFFFFFFFF



For Cortex®-R4 and Cortex®-R5 models, the range for PL111 CLCD is 0xA0000000-0xA0010000

8.4 VE interrupt assignments for Cortex-A series

The platform routes the following Shared Peripheral Interrupts (SPIs) to the GIC.

Table 8-6: SPI GIC assignments

IRQ ID	SPI offset	Device
32	0	Watchdog, SP805
34	2	Dual timer 0/1, SP804
35	3	Dual timer 2/3, SP804
36	4	Real-time Clock, PL031
37	5	UART0, PL011
38	6	UART1, PL011
39	7	UART2, PL011
40	8	UART3, PL011
41	9	MCI, PL180, MCIINTRO
42	10	MCI, PL180, MCIINTR1
43	11	AACI, PL041
44	12	KMI - Keyboard, PL050
45	13	KMI - Mouse, PL050
46	14	Color LCD Controller, PL111
47	15	Ethernet, SMSC 91C111
74	42	Virtio block device
75	43	Virtio P9 device

IRQ ID	SPI offset	Device
92	60	CPU 0 PMU
93	61	CPU 1 PMU
94	62	CPU 2 PMU
95	63	CPU 3 PMU
117	85	HDLCD

8.5 VE interrupt assignments for Cortex-R series

These tables describe the interrupt assignments for Cortex-R4, Cortex-R5, Cortex-R7 and Cortex-R8.

Table 8-7: Interrupt assignments for Cortex-R4 and Cortex-R5

GIC IRQ ID	SPI offset	Device
32	0	Watchdog, SP805
34	2	Dual timer 0/1, SP804
35	3	Dual timer 2/3, SP804
36	4	Real-time Clock, PL031
41	9	MCI, PL180, MCIINTRO
42	10	MCI, PL180, MCIINTR1
43	11	AACI, PL041
44	12	KMI - Keyboard, PL050
45	13	KMI - Mouse, PL050
47	15	Ethernet, SMSC 91C111
75	16	Level 2 Cache Controller, PL310 Combined interrupt
76	14	Color LCD Controller, PL111
96	5	UART0, PL011
97	6	UART1, PL011
98	7	UART2, PL011
99	8	UART3, PL011

Table 8-8: Interrupt assignments for Cortex-R7 and Cortex-R8

GIC IRQ ID	SPI offset	Device
32	0	Watchdog, SP805
34	2	Dual timer 0/1, SP804
35	3	Dual timer 2/3, SP804
36	4	Real-time Clock, PL031
37	5	UART0, PL011
38	6	UART1, PL011
39	7	UART2, PL011
40	8	UART3, PL011

GIC IRQ ID	SPI offset	Device
41	9	MCI, PL180, MCIINTR0
42	10	MCI, PL180, MCIINTR1
43	11	AACI, PL041
44	12	KMI - Keyboard, PL050
45	13	KMI - Mouse, PL050
46	14	Color LCD Controller, PL111
47	15	Ethernet, SMSC 91C111

8.6 VE parameters

This section describes the VE FVP instantiation parameters.

8.6.1 VE instantiation parameters

This table describes the instantiation parameters for VE models.

Table 8-9: VE instantiation parameters

Component	Parameter	Type	Allowed values	Default value	Description
ve_sysregs	user_switches_value	Integer	See VE switch S6 .	0	Switch S6 setting.
flashloader0	fname	String	Valid filename	""	Path to flash image file.
flashloader1	fname	String	Valid filename	""	Path to flash image file.
mmc	p_mmc_file	String	Valid filename	mmc.dat	Multimedia card filename.
pl111_clcd	pixel_double_limit	Integer	-	12c	Sets threshold in horizontal pixels below which pixels sent to framebuffer are doubled in size in both dimensions.
sp810_sysctrl	use_s8	Boolean	true, false	false	Indicates whether to read boot_switches_value.

8.6.2 VE secure memory parameters

This table describes the VE FVP secure memory parameters that you can change when you start the model.

Table 8-10: VE secure memory parameters

Name	Type	Allowed values	Default value	Description
daughterboard.secure_memory	Boolean	true, false	false	<p>false</p> <p>The platform behaves as before.</p> <p>true</p> <p>The address space is segregated according to the security mode of the core. Some memory blocks near the bottom of the address space are available to Secure transactions only. The rest of the address space is available to Non-secure transactions only.</p>

8.6.3 VE switch S6

This section describes the behavior and default positions of the VE system model switch.

Switch S6 is equivalent to the Boot Monitor configuration switch on the VE hardware.

If you have the standard Arm® Boot Monitor flash image loaded, the setting of switch S6-1 changes what happens on model reset. Otherwise, the function of switch S6 is implementation dependent.

To write the switch position directly to the S6 parameter in the model, you must convert the switch settings to an integer value from the equivalent binary, where 1 is on and 0 is off.

Table 8-11: Default positions of VE system model switch

Switch	Default position	Function in default position
S6-1	OFF	Displays prompt permitting Boot Monitor command entry after system start.
S6-2	OFF	See STDIO redirection, below.
S6-3	OFF	See STDIO redirection, below.
S6-4 to S6-8	OFF	Reserved for application use.

If S6-1 is in the ON position, the Boot Monitor executes the boot script that was loaded into flash. If there is no script, the Boot Monitor prompt is displayed.

The settings of S6-2 and S6-3 affect STDIO source and destination on model reset.

Table 8-12: STDIO redirection

S6-2	S6-3	Output	Input	Description
OFF	OFF	UART0	UART0	STDIO autodetects whether to use semihosting I/O or a UART. If a debugger is connected, STDIO is redirected to the debugger output window, otherwise STDIO goes to UART0.
OFF	ON	UART0	UART0	STDIO is redirected to UART0, regardless of semihosting settings.

S6-2	S6-3	Output	Input	Description
ON	OFF	CLCD	Keyboard	STDIO is redirected to the CLCD and keyboard, regardless of semihosting settings.
ON	ON	CLCD	UART0	STDIO output is redirected to the LCD and input is redirected to UART0, regardless of semihosting settings.

8.7 VE components

A complete model implementation of the VE platform includes both VE-specific components and generic components, such as buses and timers.

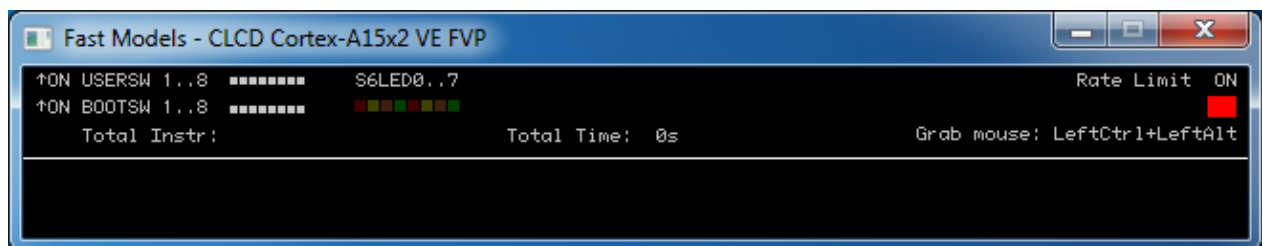
To see a list of all the component instances in the model, run it with the `--list-instances` option.

The generic components are documented in Fast Models components.

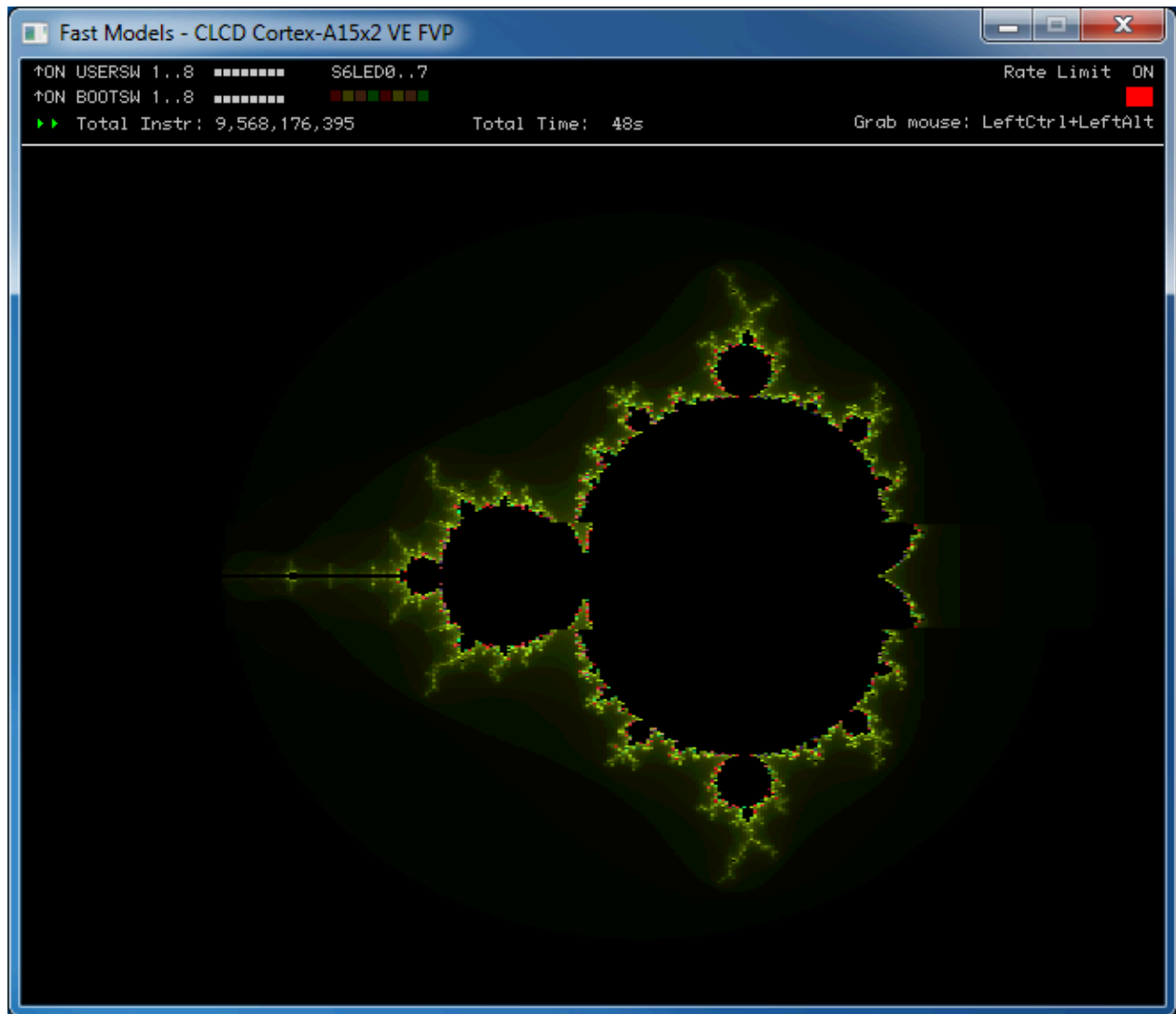
8.7.1 VEVisualisation component

This component can generate events from the host mouse or keyboard when the visualization window is in focus. For example, you can toggle the switch elements from the visualization window.

Figure 8-2: VE FVP CLCD visualization window at startup



When a suitable application or system image loads, and configures the PL111_CLCD controller registers, the window expands to show the contents of the framebuffer.

Figure 8-3: VE FVP CLCD showing contents of framebuffer

The VEVisualisation LISA+ component can be found in the `$PVLIB_HOME/examples/LISA/FVP_VE/LISA/` directory.



Note

Using this component can reduce simulation performance. Use the `rate_limit-enable` parameter to control simulation speed.

8.7.1.1 VEEVisualisation ports

This section describes the VEEVisualisation component ports.

Table 8-13: VEEVisualisation ports

Name	Protocol	Type	Description
boot_switch	ValueState	Slave	Provides state for the eight Boot DIP switches on the right side of the CLCD status bar.
clock_50Hz	ClockSignal	Slave	50Hz clock input.
cluster0_ticks[4]	InstructionCount	Slave	Connection from processor model in cluster 0 to show its current instruction count.
cluster1_ticks[4]	InstructionCount	Slave	Connection from processor model in cluster 1 to show its current instruction count.
daughter_leds	ValueState	Slave	A read/write port to read and set the value of the LEDs. 1 bit per LED, LSB left-most, up to 32 LEDs available. The LEDs appear only when parameter <code>daughter_led_count</code> is set to nonzero.
daughter_user_switches	ValueState	Slave	A read port to return the value of the daughter user switches. Write to this port to set the value of the switches, and use during reset only. LSB is left-most, up to 32 switches available.
keyboard	KeyboardStatus	Master	Output port providing key change events when the visualization window is in focus.
lcd	LCD	Slave	Connection from a CLCD controller for visualization of the frame buffer.
lcd_layout	LCDLayoutInfo	Master	Layout information for alphanumeric LCD display.
leds	ValueState	Slave	Displays state using the eight colored LEDs on the status bar.
mouse	MouseStatus	Master	Output port providing mouse movement and button events when the visualization window is in focus.
touch_screen	MouseStatus	Master	Provides mouse events when the visualization window is in focus.
user_switches	ValueState	Slave	Provides state for the eight User DIP switches on the left side of the CLCD status bar, equivalent to switch S6 on VE hardware.

8.7.1.2 VEEVisualisation parameters

This section describes the configuration parameters.

The syntax to use in a configuration file or on the command line is:

```
motherboard.vis.<parameter>=<value>
```



Setting the `rate_limit-enable` parameter to `true` prevents the simulation from running too fast on fast workstations and enables timing loops and mouse actions to work correctly. However, it reduces the overall simulation speed. If your priority is high simulation speed, set `rate_limit-enable` to `false`, which is the default.

Table 8-14: VEEVisualisation parameters

Name	Type	Allowed values	Default value	Description
cluster0_name	string	-	Cluster0	Label for cluster 0 performance values.

Name	Type	Allowed values	Default value	Description
cluster1_name	string	-	Cluster1	Label for cluster 1 performance values.
cluster2_name	string	-	Cluster2	Label for cluster 2 performance values.
cluster3_name	string	-	Cluster3	Label for cluster 3 performance values.
cpu_name	string	-	-	Processor name displayed in window title.
daughter_led_count	int	0-32	0	Set to nonzero to display up to 32 LEDs. See the <code>daughter_leds</code> port.
daughter_user_switch_count	int	0-32	0	Set this parameter to display up to 32 switches. See the <code>daughter_user_switches</code> port.
diagnostics	bool	true, false	false	Print diagnostic messages.
disable_visualisation	bool	true, false	false	Disable the VEVisualisation component on model startup.
rate_limit-enable	bool	true, false	false	Restrict simulation speed so that simulation time more closely matches real time rather than running as fast as possible.
recorder.checkInstructionCount	bool	true, false	true	Check instruction count in recording file against actual instruction count during playback.
recorder.playbackFileName	string	-	""	Playback filename (empty string disables playback).
recorder.recordingFileName	string	-	""	Recording filename (empty string disables recording).
recorder.recordingTimeBase	int	-	0x5F5E100	Timebase in 1/s (relative to the master clock where 100000000 means 10 nanoseconds resolution simulated time for a 1Hz master clock) for recording. Higher values give higher time resolution, playback timebase is always taken from the playback file.
recorder.verbose	int	-	0	Enable verbose messages (1=normal, 2=even more).
trap_key	int	Valid ATKeyCode key value. See <code>\$PVLIB_HOME/include/components/KeyCode.h</code> for a list of values.	0x4A	Trap key that works with left Ctrl key to toggle mouse display. The default is the left Alt key, so press Left Alt and Left Ctrl simultaneously to toggle the mouse display.
window_title	string	-	"Fast Models - CLCD %cpu%"	Window title. <code>cpu_name</code> replaces <code>%cpu%</code> .

8.7.1.3 VEVisualisation verification and testing

This component passes tests by use as an I/O device for booting Linux and other operating systems.

8.7.1.4 VEEVisualisation performance

Arm expects the elements in the status bar to have little effect on the performance of PV systems. However, applications that often redraw the contents of the frame buffer might incur overhead through GUI interactions on the host OS.

8.7.1.5 VEEVisualisation library dependencies

This component relies on the Simple DirectMedia Layer (SDL) libraries, specifically `libSDL2-2.0.so.0.10.0`.

This library is bundled with Fast Models and is also available as an rpm for Red Hat Enterprise Linux. On Windows, the library is called `SDL2.dll`.

Related information

[Simple DirectMedia Layer \(SDL\) cross-platform development library](#)

8.8 VE_SysRegs component

This section describes the VE system registers component.

8.8.1 VE_SysRegs

This LISA+ component is a model of the VE status and system control registers.

8.8.2 VE_SysRegs ports

This table describes the VE_SysRegs ports.

Table 8-15: VE_SysRegs ports

Name	Protocol	Type	Description
<code>cb[0-1]</code>	<code>VECBProtocol</code>	Master	The Configuration Bus (CB) controls the power and reset sequence.
<code>clock_24Mhz</code>	<code>ClockSignal</code>	Slave	Reference clock for internal counter register.
<code>clock_100Hz</code>	<code>ClockSignal</code>	Slave	Reference clock for internal counter register.
<code>clock_CLCD</code>	<code>ClockRateControl</code>	Master	The clock for the LCD controller.
<code>lcd</code>	<code>LCD</code>	Master	Multimedia bus interface output to the LCD.
<code>leds</code>	<code>ValueState</code>	Master	Displays state of the <code>SYS_LED</code> register using the eight colored LEDs on the status bar.
<code>mmb[0-2]</code>	<code>LCD</code>	Slave	Multimedia bus interface input.
<code>mmc_card_present</code>	<code>StateSignal</code>	Slave	Indicates the presence of a MultiMedia Card (MMC) image.
<code>pvbuss</code>	<code>PVBus</code>	Slave	Slave port for connection to PV bus master/decoder.

Name	Protocol	Type	Description
system_reset	Signal	Master	Signal to the platform a complete system reset. Writes to the System Configuration registers can trigger the reset signal.
user_switches	ValueState	Master	Provides state for the eight User DIP switches on the left side of the CLCD status bar, equivalent to switch S6 on VE hardware.

8.8.3 VE_SysRegs parameters

This table describes the VE_SysRegs parameters.

Table 8-16: VE_SysRegs parameters

Name	Type	Default value	Description
exit_on_shutdown	bool	false	Used to shut down the system. See the Note after the table.
mmbSiteDefault	int	1	Default MultiMedia Bus (MMB) source (0=motherboard, 1=daughterboard 1, 2=daughterboard 2).
sys_proc_id0	int	0x0c000000	Processor ID register at CoreTile Express Site 1.
sys_proc_id1	int	0xff000000	Processor ID at CoreTile Express Site 2.
tilePresent	bool	true	Tile fitted.
user_switches_value	int	0	User switches.



When `exit_on_shutdown` is `true`, if software uses the `sys_cfgctrl` function `sys_cfg_shutdown`, then the simulator shuts down and exits. For more information on the `sys_cfgctrl` function values, see the Motherboard Express µATX V2M-P1 Technical Reference Manual.

8.8.4 VE_SysRegs registers

This table describes the configuration registers.

Table 8-17: VE_SysRegs registers

Name	Offset	Access	Description
SYS_ID	0x00	Read/write	System identity
SYS_SW	0x04	Read/write	Bits[7:0] map to switch S6
SYS_LED	0x08	Read/write	Bits[7:0] map to user LEDs
SYS_100HZ	0x24	Read only	100Hz counter
SYS_FLAGS	0x30	Read/write	General purpose flags
SYS_FLAGSCLR	0x34	Write only	Clear bits in general purpose flags
SYS_NVFLAGS	0x38	Read/write	General purpose non-volatile flags
SYS_NVFLAGSCLR	0x3C	Write only	Clear bits in general purpose non-volatile flags
SYS_MCI	0x48	Read only	MCI
SYS_FLASH	0x4C	Read/write	Flash control

Name	Offset	Access	Description
SYS_CFGSW	0x58	Read/write	Boot select switch
SYS_24MHZ	0x5C	Read only	24MHz counter
SYS_MISC	0x60	Read/write	Miscellaneous control flags
SYS_DMA	0x64	Read/write	DMA peripheral map
SYS_PROCID0	0x84	Read/write	Processor ID
SYS_PROCID1	0x88	Read/write	Processor ID
SYS_CFGDATA	0xA0	Read/write	Data to be read/written from/to motherboard controller
SYS_CFGCTRL	0xA4	Read/write	Control data transfer to motherboard controller
SYS_CFGSTAT	0xA8	Read/write	Status of data transfer to motherboard

8.8.5 VE_SysRegs - verification and testing

This component was tested as part of the Versatile™ Express model.

8.9 Differences between the VE hardware and the system models

This section describes features of the hardware that the models do not implement, or implement with significant differences.

Related information

[Fast Models accuracy](#) on page 22

[Processor implementation](#) on page 27

8.9.1 Memory map

The model represents the memory map of the hardware VE platform, but is not an accurate representation of a specific revision.

The memory map in the supplied model is sufficiently complete and accurate to boot the same operating system images as for the VE hardware.

In the memory map, memory regions that are not explicitly occupied by a peripheral or by memory are unmapped. This includes regions otherwise occupied by a peripheral that is not implemented, and those areas that are documented as reserved. Accessing these regions from the host processor results in the model presenting a warning.

8.9.2 Memory aliasing

The model implements address-space aliasing of the DRAM. This means that the same physical memory locations are visible at different addresses.

The lower 2GB of the DRAM is accessible at 0x00_80000000. The full 4GB of DRAM is accessible at 0x08_00000000 and again at 0x80_00000000. The aliasing of DRAM then repeats from 0x81_00000000 up to 0xFF_FFFFFFFF.

8.9.3 VE hardware features absent

These FVPs do not implement the following features of the hardware:

- Two-wire serial bus interfaces.
- USB interfaces.
- PCI Express interfaces.
- Compact flash.
- Digital Visual Interface (DVI).
- Debug and test interfaces.
- Dynamic Memory Controller (DMC).
- Static Memory Controller (SMC).

Related information

[VE memory map for Cortex-A_series](#) on page 4261

8.9.4 VE hardware features different

These Fixed Virtual Platforms only partially implement some features of the hardware.

The partially implemented features might not work as you expect. Check the model release notes for the latest information.

Sound

The VE FVPs implement the PL041 AACI PrimeCell and the audio CODEC as in the VE hardware, but with a limited number of sample rates.

Proprietary Notice

This document is protected by copyright and other related rights and the use or implementation of the information contained in this document may be protected by one or more patents or pending patent applications. No part of this document may be reproduced in any form by any means without the express prior written permission of Arm Limited ("Arm"). No license, express or implied, by estoppel or otherwise to any intellectual property rights is granted by this document unless specifically stated.

Your access to the information in this document is conditional upon your acceptance that you will not use or permit others to use the information for the purposes of determining whether the subject matter of this document infringes any third party patents.

The content of this document is informational only. Any solutions presented herein are subject to changing conditions, information, scope, and data. This document was produced using reasonable efforts based on information available as of the date of issue of this document. The scope of information in this document may exceed that which Arm is required to provide, and such additional information is merely intended to further assist the recipient and does not represent Arm's view of the scope of its obligations. You acknowledge and agree that you possess the necessary expertise in system security and functional safety and that you shall be solely responsible for compliance with all legal, regulatory, safety and security related requirements concerning your products, notwithstanding any information or support that may be provided by Arm herein. conjunction with any Arm technology described in this document, and to minimize risks, adequate design and operating safeguards should be provided for by you.

This document may include technical inaccuracies or typographical errors. THIS DOCUMENT IS PROVIDED "AS IS". ARM PROVIDES NO REPRESENTATIONS AND NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NON-INFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO THE DOCUMENT. For the avoidance of doubt, Arm makes no representation with respect to, and has undertaken no analysis to identify or understand the scope and content of, any patents, copyrights, trade secrets, trademarks, or other rights.

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL ARM BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Reference by Arm to any third party's products or services within this document is not an express or implied approval or endorsement of the use thereof.

This document consists solely of commercial items. You shall be responsible for ensuring that any permitted use, duplication, or disclosure of this document complies fully with any relevant export laws and regulations to assure that this document or any portion thereof is not exported,

directly or indirectly, in violation of such export laws. Use of the word “partner” in reference to Arm’s customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of this document shall prevail.

The validity, construction and performance of this notice shall be governed by English Law.

The Arm corporate logo and words marked with ® or ™ are registered trademarks or trademarks of Arm Limited (or its affiliates) in the US and/or elsewhere. Please follow Arm’s trademark usage guidelines at <https://www.arm.com/company/policies/trademarks>. All rights reserved. Other brands and names mentioned in this document may be the trademarks of their respective owners.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

PRE-1121-V1.0

Product and document information

Read the information in these sections to understand the release status of the product and documentation, and the conventions used in Arm documents.

Product status

All products and services provided by Arm require deliverables to be prepared and made available at different levels of completeness. The information in this document indicates the appropriate level of completeness for the associated deliverables.

Product completeness status

The information in this document is Final, that is for a developed product.

Revision history

These sections can help you understand how the document has changed over time.

Document release information

The Document history table gives the issue number and the released date for each released issue of this document.

Document history

Issue	Date	Confidentiality	Change
1126-00	19 June 2024	Non-Confidential	Update for v11.26
1125-00	13 March 2024	Non-Confidential	Update for v11.25.
1124-00	6 December 2023	Non-Confidential	Update for v11.24.
1123-00	13 September 2023	Non-Confidential	Update for v11.23.
1122-00	14 June 2023	Non-Confidential	Update for v11.22.
1121-00	22 March 2023	Non-Confidential	Update for v11.21.

Issue	Date	Confidentiality	Change
1120-00	7 December 2022	Non-Confidential	Update for v11.20.
1119-00	14 September 2022	Non-Confidential	Update for v11.19.
1118-00	15 June 2022	Non-Confidential	Update for v11.18.
1117-00	16 February 2022	Non-Confidential	Update for v11.17.
1116-00	6 October 2021	Non-Confidential	Update for v11.16.
1115-00	29 June 2021	Non-Confidential	Update for v11.15.
1114-01	30 March 2021	Non-Confidential	Document update 1 for v11.14.
1114-00	17 March 2021	Non-Confidential	Update for v11.14.
1113-00	9 December 2020	Non-Confidential	Update for v11.13.
1112-00	22 September 2020	Non-Confidential	Update for v11.12.
1111-00	9 June 2020	Non-Confidential	Update for v11.11.
1110-00	12 March 2020	Non-Confidential	Update for v11.10.
1109-00	28 November 2019	Non-Confidential	Update for v11.9.
1108-00	5 September 2019	Non-Confidential	Update for v11.8.
1107-00	17 May 2019	Non-Confidential	Update for v11.7.
1106-01	25 March 2019	Non-Confidential	Update for v11.6.1.
1106-00	27 February 2019	Non-Confidential	Update for v11.6.

Issue	Date	Confidentiality	Change
1105-00	23 November 2018	Non-Confidential	Update for v11.5.
1104-01	17 August 2018	Non-Confidential	Update for v11.4.2.
1104-00	22 June 2018	Non-Confidential	Update for v11.4.
1103-00	23 February 2018	Non-Confidential	Update for v11.3.
1102-00	17 November 2017	Non-Confidential	Update for v11.2.
1101-00	31 August 2017	Non-Confidential	Update for v11.1.
1100-00	31 May 2017	Non-Confidential	Update for v11.0. Document numbering scheme has changed.

Change history

For technical changes to this documentation, see the [Fast Models Release Notes](#).

Conventions

The following subsections describe conventions used in Arm documents.

Glossary

The Arm Glossary is a list of terms used in Arm documentation, together with definitions for those terms. The Arm Glossary does not contain terms that are industry standard unless the Arm meaning differs from the generally accepted meaning.

See the Arm Glossary for more information: developer.arm.com/glossary.

Typographic conventions

Arm documentation uses typographical conventions to convey specific meaning.

Convention	Use
<i>italic</i>	Citations.
bold	Interface elements, such as menu names. Terms in descriptive lists, where appropriate.
<code>monospace</code>	Text that you can enter at the keyboard, such as commands, file and program names, and source code.

Convention	Use
monospace <u>underline</u>	A permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.
<and>	Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example: <div>MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2></div>
SMALL CAPITALS	Terms that have specific technical meanings as defined in the <i>Arm® Glossary</i> . For example, IMPLEMENTATION DEFINED , IMPLEMENTATION SPECIFIC , UNKNOWN , and UNPREDICTABLE .



We recommend the following. If you do not follow these recommendations your system might not work.



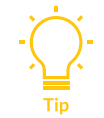
Your system requires the following. If you do not follow these requirements your system will not work.



You are at risk of causing permanent damage to your system or your equipment, or harming yourself.



This information is important and needs your attention.



A useful tip that might make it easier, better or faster to perform a task.



A reminder of something important that relates to the information you are reading.

Useful resources

This document contains information that is specific to this product. See the following resources for other useful information.

Access to Arm documents depends on their confidentiality:

- Non-Confidential documents are available at developer.arm.com/documentation. Each document link in the following tables goes to the online version of the document.
- Confidential documents are available to licensees only through the product package.

Arm product resources	Document ID	Confidentiality
A-Profile Architecture	–	Non-Confidential
AMBA-PV Extensions to TLM User Guide	100962	Non-Confidential
AN521 Example SSE-200 Subsystem for MPS2+ Application Note	DAI 0521	Non-Confidential
Arm® Architecture Models	–	Non-Confidential
Arm® Corstone™ SSE-300 Example Subsystem Technical Reference Manual	101773	Non-Confidential
Arm® Cortex®-M23 Armv8-M IoT Kit User Guide	ECM 0635473	Non-Confidential
Arm® Cortex®-M33 processor Armv8-M IoT Kit FVP User Guide	ECM 0601256	Non-Confidential
Arm® Cortex®-M7 SMM on V2M-MPS2 Application Note 400	DAI 0400	Non-Confidential
Arm® DynamIQ Shared Unit Technical Reference Manual	100453	Non-Confidential
Arm® Socrates™	–	Non-Confidential
ASTFplugin FAQs	108074	Non-Confidential
Component Architecture Debug Interface User Guide	100963	Non-Confidential
Fast Models Fixed Virtual Platforms (FVP) Reference Guide	100966	Non-Confidential
Fast Models Model Trace Interface Reference Manual	DUI 0819	Non-Confidential
Fast Models User Guide	100965	Non-Confidential
Fixed Virtual Platforms	–	Non-Confidential
How to generate ASTF traces of workloads running on Fast Models	109193	Non-Confidential
Introduction to SVE2	102340	Non-Confidential
LISA+ Language for Fast Models Reference Guide	101092	Non-Confidential
Model Debugger for Fast Models User Guide	100968	Non-Confidential
Motherboard Express µATX V2M-P1 Technical Reference Manual	DUI 0447	Non-Confidential
Product Download Hub	–	Non-Confidential
Workload Trace Generation Best Practices	107983	Non-Confidential

Arm architecture and specifications	Document ID	Confidentiality
AMBA® Low Power Interface Specification	IHI 0068	Non-Confidential
Armv7-M Architecture Reference Manual	DDI 0403	Non-Confidential
Arm® Architecture Reference Manual Armv7-A and Armv7-R edition	DDI 0406	Non-Confidential
Arm® Architecture Reference Manual for A-profile architecture	DDI 0487	Non-Confidential

Arm architecture and specifications	Document ID	Confidentiality
<i>Arm® Architecture Reference Manual Supplement, The Scalable Vector Extension</i>	DDI 0584	Non-Confidential
<i>Arm® Generic Interrupt Controller Architecture version 2.0 - Architecture Specification</i>	IHI 0048	Non-Confidential
<i>Arm® Power Policy Unit Architecture Specification</i>	DEN 0051	Non-Confidential
<i>Arm® Realm Management Extension (RME) System Architecture</i>	DEN 0129	Non-Confidential

Non-Arm resources	Document ID	Organization
<i>Accellera Systems Initiative</i>	–	http://www.accellera.org
<i>GitHub</i>	–	Github
<i>Intel Download Center, Intel StrataFlash Memory (J3) datasheet</i>	–	Intel
<i>MultiMedia Card Association specification</i>	–	JEDEC
<i>Simple DirectMedia Layer Cross-platform Development Library</i>	–	Simple DirectMedia Layer
<i>Virtual I/O Device (VIRTIO) Version 1.0</i>	–	OASIS Open