# OVP Guide to Using Processor Models

# Model specific information for
# ARM_Cortex-R82MPx5

## Imperas Software Limited
Imperas Buildings, North Weston
Thame, Oxfordshire, OX9 2HA, U.K.
docs@imperas.com

## Copyright Notice

Copyright (c) 2024 Imperas Software Limited. All rights reserved. This software and documentation contain information that is the property of Imperas Software Limited. The software and documentation are furnished under a license agreement and may be used or copied only in accordance with the terms of the license agreement. No part of the software and documentation may be reproduced, transmitted, or translated, in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without prior written permission of Imperas Software Limited, or as expressly provided by the license agreement.

## Right to Copy Documentation

The license agreement with Imperas permits licensee to make copies of the documentation for its internal use only. Each copy shall include all copyrights, trademarks, service marks, and proprietary rights notices, if any.

## Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the readers responsibility to determine the applicable regulations and to comply with them.

## Disclaimer

IMPERAS SOFTWARE LIMITED, AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

## Model Release Status

This model is released as part of OVP releases and is included in OVPworld packages. Please visit OVPworld.org.

# Contents

# Chapter 1

# Overview

This document provides the details of an OVP Fast Processor Model variant.

OVP Fast Processor Models are written in C and provide a C API for use in C based platforms. The models also provide a native interface for use in SystemC TLM2 platforms.

The models are written using the OVP VMI API that provides a Virtual Machine Interface that defines the behavior of the processor. The VMI API makes a clear line between model and simulator allowing very good optimization and world class high speed performance. Most models are provided as a binary shared object and also as source. This allows the download and use of the model binary or the use of the source to explore and modify the model.

The models are run through an extensive QA and regression testing process and most model families are validated using technology provided by the processor IP owners. There is a companion document (OVP Guide to Using Processor Models) which explains the general concepts of OVP Fast Processor Models and their use. It is downloadable from the OVPworld website documentation pages.

## 1.1 Description

ARM Processor Model

## 1.2 Licensing

Usage of binary model under license governing simulator usage.

Note that for models of ARM CPUs the license includes the following terms:

Licensee is granted a non-exclusive, worldwide, non-transferable, revocable licence to:

If no source is being provided to the Licensee: use and copy only (no modifications rights are granted) the model for the sole purpose of designing, developing, analyzing, debugging, testing, verifying, validating and optimizing software which: (a) (i) is for ARM based systems; and (ii) does not incorporate the ARM Models or any part thereof; and (b) such ARM Models may not be used

to emulate an ARM based system to run application software in a production or live environment.

If source code is being provided to the Licensee: use, copy and modify the model for the sole purpose of designing, developing, analyzing, debugging, testing, verifying, validating and optimizing software which: (a) (i) is for ARM based systems; and (ii) does not incorporate the ARM Models or any part thereof; and (b) such ARM Models may not be used to emulate an ARM based system to run application software in a production or live environment.

In the case of any Licensee who is either or both an academic or educational institution the purposes shall be limited to internal use.

Except to the extent that such activity is permitted by applicable law, Licensee shall not reverse engineer, decompile, or disassemble this model. If this model was provided to Licensee in Europe, Licensee shall not reverse engineer, decompile or disassemble the Model for the purposes of error correction.

The License agreement does not entitle Licensee to manufacture in silicon any product based on this model.

The License agreement does not entitle Licensee to use this model for evaluating the validity of any ARM patent.

Source of model available under separate Imperas Software License Agreement.

## 1.3   Limitations

Instruction pipelines are not modeled in any way. All instructions are assumed to complete immediately. This means that instruction barrier instructions (e.g. ISB, CP15ISB) are treated as NOPs, with the exception of any undefined instruction behavior, which is modeled. The model does not implement speculative fetch behavior. The branch cache is not modeled.

Caches and write buffers are not modeled in any way. All loads, fetches and stores complete immediately and in order, and are fully synchronous (as if the memory was of Strongly Ordered or Device-nGnRnE type). Data barrier instructions (e.g. DSB, CP15DSB) are treated as NOPs, with the exception of any undefined instruction behavior, which is modeled. Cache manipulation instructions are implemented as NOPs, with the exception of any undefined instruction behavior, which is modeled.

Real-world timing effects are not modeled: all instructions are assumed to complete in a single cycle.

Performance Monitors are implemented as a register interface only except for the cycle counter, which is implemented assuming one instruction per cycle.

Self-hosted Trace Extension registers are implemented as a register interface only.

Debug registers are implemented but non-functional (which is sufficient to allow operating systems such as Linux to boot). Debug state is not implemented.

The GICv3 block is implemented without any ITS. Implementation-defined GICR registers for control of LPIs (GICR_SETLPIR, GICR_CLRLPIR, GICR_INVLPIR, GICR_INVALLR and GICR_SYNCR) are all implemented.

Copyright (c) 2024 Imperas Software Limited      www.ovpworld.org

OVP License. Release 20240902.0      Page 2 of 56

The optional ITCM region is not implemented.

The optional DTCM region is not implemented.

The optional LLP region is not implemented.

The optional LLRAM region is not implemented.

The optional SPP region is not implemented.

## 1.4    Verification

Models have been extensively tested by Imperas. ARM Cortex models have been successfully used by customers to simulate SMP Linux, Ubuntu Desktop, VxWorks and ThreadX on Xilinx Zynq virtual platforms.

## 1.5    Features

The precise set of implemented features in the model is defined by ID registers. Use overrides to modify these if required (for example override_PFR0 or override_AA64PFR0_EL1).

### 1.5.1    Core Features

AArch64 is implemented at EL2, EL1 and EL0.

The following ARMv8.0 core features are implemented: FEAT_CSV2_1p1, FEAT_CSV3, FEAT_DGH, FEAT_SB, FEAT_SPECRES, FEAT_SSBS, FEAT_SSBS2.

The following ARMv8.1 core features are implemented: FEAT_LSE, FEAT_PAN, FEAT_RDM.

The following ARMv8.2 core features are implemented: FEAT_DotProd, FEAT_DPB, FEAT_DPB2, FEAT_FHM, FEAT_FP16, FEAT_IESB, FEAT_PAN2, FEAT_RAS (with no error records), FEAT_UAO, FEAT_XNX.

The following ARMv8.3 core features are implemented: FEAT_LRCPC, FEAT_FCMA, FEAT_FPAC, FEAT_JSCVT, FEAT_PAuth, FEAT_PAuth2.

The following ARMv8.4 core features are implemented: FEAT_DIT, FEAT_FlagM, FEAT_IDST, FEAT_LRCPC2, FEAT_LSE2, FEAT_RASv1p1, FEAT_S2FWB, FEAT_SEL2, FEAT_TLBIOS, FEAT_TLBIRANGE.

### 1.5.2    Memory System

Security extensions are implemented (also known as TrustZone). To make non-secure accesses visible externally, override ID_AA64MMFR0_EL1.PARange to specify the required physical bus size (32, 36, 40, 42, 44, 48 or 52 bits) and connect the processor to a bus one bit wider (33, 37, 41, 43, 45, 49 or 53 bits, respectively). The extra most-significant bit is the NS bit, indicating a

non-secure access. If non-secure accesses are not required to be made visible externally, connect the processor to a bus of exactly the size implied by ID_AA64MMFR0_EL1.PARange.

LPA (large physical address extension) is implemented as standard in ARMv8.

### 1.5.3 Advanced SIMD and Floating-Point Features

SIMD and VFP instructions are implemented.

The model implements trapped exceptions if FPTrap is set to 1 in MVFR0 (for AArch32) or MVFR0_EL1 (for AArch64). When floating point exception traps are taken, cumulative exception flags are not updated (in other words, cumulative flag state is always the same as prior to instruction execution, even for SIMD instructions). When multiple enabled exceptions are raised by a single floating point operation, the exception reported is the one in least-significant bit position in FPSCR (for AArch32) or FPCR (for AArch64). When multiple enabled exceptions are raised by different SIMD element computations, the exception reported is selected from the lowest-index-number SIMD operation. Contact Imperas if requirements for exception reporting differ from these.

Trapped exceptions not are implemented in this variant (FPTrap=0)

### 1.5.4 Generic Timer

Generic Timer is present. Use parameter "override_timerScaleFactor" to specify the counter rate as a fraction of the processor MIPS rate (e.g. 10 implies Generic Timer counters increment once every 10 processor instructions).

### 1.5.5 Generic Interrupt Controller

GIC block is implemented (GICv3, including security extensions). Accesses to GIC registers can be viewed externally by connecting to the 32-bit GICRegisters and GICDRegisters bus ports. Secure register accesses are at offset 0x0 on these busses; for example, a secure access to GICD register GICD_CTLR can be observed by monitoring address 0x00000000 of bus GICDRegisters. Non-secure accesses are at offset 0x80000000 on these busses; for example, a non-secure access to GICD register GICD_CTLR can be observed by monitoring address 0x80000000 of bus GICDRegisters

GIC Distributor registers are located at address 0x2f000000. Use parameter "override_GICv3_DistributorBase" to change this if required.

The internal GIC block can be disabled by raising signal GICCDISABLE, in which case the GIC needs to be modeled using a platform component instead. Input signals vfiq_CPU<N>and virq_CPU<N>can be used by this component to raise virtual FIQ and IRQ interrupts on cores in the cluster if required.

## 1.6 Debug Mask

It is possible to enable model debug features in various categories. This can be done statically using the "override_debugMask" parameter, or dynamically using the "debugflags" command. Enabled

Copyright (c) 2024 Imperas Software Limited     www.ovpworld.org

OVP License. Release 20240902.0     Page 4 of 56

debug features are specified using a bitmask value, as follows:

Value 0x004: enable debugging of MMU/MPU mappings.

Value 0x020: enable debugging of reads and writes of GIC block registers.

Value 0x040: enable debugging of exception routing via the GIC model component.

Value 0x080: enable debugging of all system register accesses.

Value 0x100: enable debugging of all traps of system register accesses.

Value 0x200: enable verbose debugging of other miscellaneous behavior (for example, the reason why a particular instruction is undefined).

Value 0x400: enable debugging of Performance Monitor timers

All other bits in the debug bitmask are reserved and must not be set to non-zero values.

## 1.7   Integration Support

This model implements a number of non-architectural pseudo-registers and other features to facilitate integration.

### 1.7.1   Halt Reason Introspection

An artifact register HaltReason can be read to determine the reason or reasons that a processor is halted. This register is a bitfield, with the following encoding: bit 0 indicates the processor has executed a wait-for-event (WFE) instruction; bit 1 indicates the processor has executed a wait-for-interrupt (WFI) instruction; and bit 2 indicates the processor is held in reset.

### 1.7.2   System Register Access Monitor

If parameter "enableSystemMonitorBus" is True, an artifact 32-bit bus "SystemMonitor" is enabled for each PE. Every system register read or write by that PE is then visible as a read or write on this artifact bus, and can therefore be monitored using callbacks installed in the client environment (use opBusReadMonitorAdd/opBusWriteMonitorAdd or icmAddBusReadCallback-/icmAddBusWriteCallback, depending on the client API). The format of the address on the bus is as follows:

bits 31:26 - zero

bit 25 - 1 if AArch64 access, 0 if AArch32 access

bit 24 - 1 if non-secure access, 0 if secure access

bits 23:20 - CRm value

bits 19:16 - CRn value

bits 15:12 - op2 value

bits 11:8 - op1 value

bits 7:4 - op0 value (AArch64) or coprocessor number (AArch32)

bits 3:0 - zero

As an example, to view non-secure writes to writes to CNTFRQ_EL0 in AArch64 state, install a write monitor on address range 0x020e0330:0x020e0333.

### 1.7.3   System Register Implementation

If parameter "enableSystemBus" is True, an artifact 32-bit bus "System" is enabled for each PE. Slave callbacks installed on this bus can be used to implement modified system register behavior (use opBusSlaveNew or icmMapExternalMemory, depending on the client API). The format of the address on the bus is the same as for the system monitor bus, described above.

Copyright (c) 2024 Imperas Software Limited     www.ovpworld.org

OVP License. Release 20240902.0     Page 6 of 56

# Chapter 2

# Configuration

## 2.1 Location

This model's VLNV is arm.ovpworld.org/processor/arm/1.0.
The model source is usually at:
$IMPERAS_HOME/ImperasLib/source/arm.ovpworld.org/processor/arm/1.0
The model binary is usually at:
$IMPERAS_HOME/lib/$IMPERAS_ARCH/ImperasLib/arm.ovpworld.org/processor/arm/1.0

## 2.2 GDB Path

The default GDB for this model is: $IMPERAS_HOME/lib/$IMPERAS_ARCH/gdb/aarch64-none-elf-gdb.

## 2.3 Semi-Host Library

The default semi-host library file is arm.ovpworld.org/semihosting/armAngel/1.0

## 2.4 Processor Endian-ness

This is a LITTLE endian model.

## 2.5 QuantumLeap Support

This processor is qualified to run in a QuantumLeap enabled simulator.

## 2.6 Processor ELF code

ELF codes supported by this model are:0xb7 and 0x28.

# Chapter 3

# All Variants in this model

This model has these variants

| Variant | Description |
| --- | --- |
| ARMv4T | |
| ARMv4xM | |
| ARMv4 | |
| ARMv4TxM | |
| ARMv5xM | |
| ARMv5 | |
| ARMv5TxM | |
| ARMv5T | |
| ARMv5TExP | |
| ARMv5TE | |
| ARMv5TEJ | |
| ARMv6 | |
| ARMv6K | |
| ARMv6T2 | |
| ARMv6KZ | |
| ARMv7 | |
| ARM7TDMI | |
| ARM7EJ-S | |
| ARM720T | |
| ARM920T | |
| ARM922T | |
| ARM926EJ-S | |
| ARM940T | |
| ARM946E | |
| ARM966E | |
| ARM968E-S | |
| ARM1020E | |
| ARM1022E | |
| ARM1026EJ-S | |
| ARM1136J-S | |
| ARM1156T2-S | |

| | |
|---|---|
| ARM1176JZ-S | |
| Cortex-R4 | |
| Cortex-R4F | |
| Cortex-R52MPx1 | |
| Cortex-R52MPx2 | |
| Cortex-R52MPx3 | |
| Cortex-R52MPx4 | |
| Cortex-R52+MPx1 | |
| Cortex-R52+MPx2 | |
| Cortex-R52+MPx3 | |
| Cortex-R52+MPx4 | |
| Cortex-R82MPx1 | |
| Cortex-R82MPx2 | |
| Cortex-R82MPx3 | |
| Cortex-R82MPx4 | |
| Cortex-R82MPx5 | (described in this document) |
| Cortex-R82MPx6 | |
| Cortex-R82MPx7 | |
| Cortex-R82MPx8 | |
| Cortex-A5UP | |
| Cortex-A5MPx1 | |
| Cortex-A5MPx2 | |
| Cortex-A5MPx3 | |
| Cortex-A5MPx4 | |
| Cortex-A8 | |
| Cortex-A9UP | |
| Cortex-A9MPx1 | |
| Cortex-A9MPx2 | |
| Cortex-A9MPx3 | |
| Cortex-A9MPx4 | |
| Cortex-A7UP | |
| Cortex-A7MPx1 | |
| Cortex-A7MPx2 | |
| Cortex-A7MPx3 | |
| Cortex-A7MPx4 | |
| Cortex-A15UP | |
| Cortex-A15MPx1 | |
| Cortex-A15MPx2 | |
| Cortex-A15MPx3 | |
| Cortex-A15MPx4 | |
| Cortex-A17MPx1 | |
| Cortex-A17MPx2 | |
| Cortex-A17MPx3 | |
| Cortex-A17MPx4 | |
| AArch32 | |

Copyright (c) 2024 Imperas Software Limited      www.ovpworld.org

OVP License. Release 20240902.0      Page 9 of 56

| AArch64 | |
|---|---|
| Cortex-A32MPx1 | |
| Cortex-A32MPx2 | |
| Cortex-A32MPx3 | |
| Cortex-A32MPx4 | |
| Cortex-A35MPx1 | |
| Cortex-A35MPx2 | |
| Cortex-A35MPx3 | |
| Cortex-A35MPx4 | |
| Cortex-A53MPx1 | |
| Cortex-A53MPx2 | |
| Cortex-A53MPx3 | |
| Cortex-A53MPx4 | |
| Cortex-A55MPx1 | |
| Cortex-A55MPx2 | |
| Cortex-A55MPx3 | |
| Cortex-A55MPx4 | |
| Cortex-A57MPx1 | |
| Cortex-A57MPx2 | |
| Cortex-A57MPx3 | |
| Cortex-A57MPx4 | |
| Cortex-A72MPx1 | |
| Cortex-A72MPx2 | |
| Cortex-A72MPx3 | |
| Cortex-A72MPx4 | |
| Cortex-A73MPx1 | |
| Cortex-A73MPx2 | |
| Cortex-A73MPx3 | |
| Cortex-A73MPx4 | |
| Cortex-A75MPx1 | |
| Cortex-A75MPx2 | |
| Cortex-A75MPx3 | |
| Cortex-A75MPx4 | |
| MultiCluster | |

Table 3.1: All Variants in this model

# Chapter 4

# Bus Master Ports

This model has these bus master ports.

| Name | min | max | Connect? | Description |
|------|-----|-----|----------|-------------|
| INSTRUCTION | 32 | 52 | mandatory | |
| DATA | 32 | 52 | optional | |
| GICRegisters | 32 | 32 | optional | GIC memory-mapped register block |
| GICDRegisters | 32 | 32 | optional | GICD memory-mapped register block |

Table 4.1: Bus Master Ports

# Chapter 5

# Bus Slave Ports

This model has no bus slave ports.

# Chapter 6

# Net Ports

This model has these net ports.

| Name | Type | Connect? | Description |
|------|------|----------|-------------|
| SPI32 | input | optional | Shared peripheral interrupt |
| SPI33 | input | optional | Shared peripheral interrupt |
| SPI34 | input | optional | Shared peripheral interrupt |
| SPI35 | input | optional | Shared peripheral interrupt |
| SPI36 | input | optional | Shared peripheral interrupt |
| SPI37 | input | optional | Shared peripheral interrupt |
| SPI38 | input | optional | Shared peripheral interrupt |
| SPI39 | input | optional | Shared peripheral interrupt |
| SPI40 | input | optional | Shared peripheral interrupt |
| SPI41 | input | optional | Shared peripheral interrupt |
| SPI42 | input | optional | Shared peripheral interrupt |
| SPI43 | input | optional | Shared peripheral interrupt |
| SPI44 | input | optional | Shared peripheral interrupt |
| SPI45 | input | optional | Shared peripheral interrupt |
| SPI46 | input | optional | Shared peripheral interrupt |
| SPI47 | input | optional | Shared peripheral interrupt |
| SPI48 | input | optional | Shared peripheral interrupt |
| SPI49 | input | optional | Shared peripheral interrupt |
| SPI50 | input | optional | Shared peripheral interrupt |
| SPI51 | input | optional | Shared peripheral interrupt |
| SPI52 | input | optional | Shared peripheral interrupt |
| SPI53 | input | optional | Shared peripheral interrupt |
| SPI54 | input | optional | Shared peripheral interrupt |
| SPI55 | input | optional | Shared peripheral interrupt |
| SPI56 | input | optional | Shared peripheral interrupt |
| SPI57 | input | optional | Shared peripheral interrupt |
| SPI58 | input | optional | Shared peripheral interrupt |
| SPI59 | input | optional | Shared peripheral interrupt |
| SPI60 | input | optional | Shared peripheral interrupt |
| SPI61 | input | optional | Shared peripheral interrupt |
| SPI62 | input | optional | Shared peripheral interrupt |

| | | | |
|---|---|---|---|
| SPI63 | input | optional | Shared peripheral interrupt |
| SPI64 | input | optional | Shared peripheral interrupt |
| SPI65 | input | optional | Shared peripheral interrupt |
| SPI66 | input | optional | Shared peripheral interrupt |
| SPI67 | input | optional | Shared peripheral interrupt |
| SPI68 | input | optional | Shared peripheral interrupt |
| SPI69 | input | optional | Shared peripheral interrupt |
| SPI70 | input | optional | Shared peripheral interrupt |
| SPI71 | input | optional | Shared peripheral interrupt |
| SPI72 | input | optional | Shared peripheral interrupt |
| SPI73 | input | optional | Shared peripheral interrupt |
| SPI74 | input | optional | Shared peripheral interrupt |
| SPI75 | input | optional | Shared peripheral interrupt |
| SPI76 | input | optional | Shared peripheral interrupt |
| SPI77 | input | optional | Shared peripheral interrupt |
| SPI78 | input | optional | Shared peripheral interrupt |
| SPI79 | input | optional | Shared peripheral interrupt |
| SPI80 | input | optional | Shared peripheral interrupt |
| SPI81 | input | optional | Shared peripheral interrupt |
| SPI82 | input | optional | Shared peripheral interrupt |
| SPI83 | input | optional | Shared peripheral interrupt |
| SPI84 | input | optional | Shared peripheral interrupt |
| SPI85 | input | optional | Shared peripheral interrupt |
| SPI86 | input | optional | Shared peripheral interrupt |
| SPI87 | input | optional | Shared peripheral interrupt |
| SPI88 | input | optional | Shared peripheral interrupt |
| SPI89 | input | optional | Shared peripheral interrupt |
| SPI90 | input | optional | Shared peripheral interrupt |
| SPI91 | input | optional | Shared peripheral interrupt |
| SPI92 | input | optional | Shared peripheral interrupt |
| SPI93 | input | optional | Shared peripheral interrupt |
| SPI94 | input | optional | Shared peripheral interrupt |
| SPI95 | input | optional | Shared peripheral interrupt |
| SPIVector | input | optional | Shared peripheral interrupt vectorized input |
| periphReset | input | optional | Peripheral reset (active high) |
| GICCDISABLE | input | optional | GIC CPU interface logic disable (active high, sampled on rising edge of periphReset) |
| PPI16_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI17_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI18_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI19_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI20_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI21_CPU0_0 | input | optional | Private peripheral interrupt |

Copyright (c) 2024 Imperas Software Limited      www.ovpworld.org

OVP License. Release 20240902.0      Page 14 of 56

| PPI22_CPU0_0 | input | optional | Private peripheral interrupt |
|---|---|---|---|
| PPI23_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI24_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI25_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI26_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI27_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI28_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI29_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI30_CPU0_0 | input | optional | Private peripheral interrupt |
| PPI31_CPU0_0 | input | optional | Private peripheral interrupt |
| CNTVIRQ_CPU0_0 | output | optional | EL1 Virtual timer event (active high) |
| CNTPNSIRQ_CPU0_0 | output | optional | EL1 Physical timer event (active high) |
| CNTHPSIRQ_CPU0_0 | output | optional | Secure EL2 Physical timer event (active high) |
| IRQOUT_CPU0_0 | output | optional | IRQ wakeup |
| FIQOUT_CPU0_0 | output | optional | FIQ wakeup |
| CLUSTERIDAFF1 | input | optional | Configure MPIDR.Aff1 |
| CLUSTERIDAFF2 | input | optional | Configure MPIDR.Aff2 |
| CLUSTERIDAFF3 | input | optional | Configure MPIDR.Aff3 |
| RVBARADDRx_CPU0_0 | input | optional | Configure AArch64 Reset Vector Base Address at reset |
| CFGEND_CPU0_0 | input | optional | Configure exception endianness (SCTLR.EE) |
| reset_CPU0_0 | input | optional | Processor reset, active high |
| fiq_CPU0_0 | input | optional | FIQ interrupt, active high (negation of nFIQ) |
| irq_CPU0_0 | input | optional | IRQ interrupt, active high (negation of nIRQ) |
| sei_CPU0_0 | input | optional | System error interrupt, active on rising edge (negation of nSEI) |
| vfiq_CPU0_0 | input | optional | Virtual FIQ interrupt, active high (negation of nVFIQ) |
| virq_CPU0_0 | input | optional | Virtual IRQ interrupt, active high (negation of nVIRQ) |
| vsei_CPU0_0 | input | optional | Virtual system error interrupt, active on rising edge (negation of nVSEI) |
| haltReason_CPU0_0 | output | optional | Indicates why core is halted |
| AXI_SLVERR_CPU0_0 | input | optional | AXI external abort type (DECERR=0, SLVERR=1) |
| CP15SDISABLE_CPU0_0 | input | optional | CP15SDISABLE (active high) |
| PMUIRQ_CPU0_0 | output | optional | Performance monitor event (active high) |
| syncCluster_CPU0_0 | output | optional | Cluster synchronization required |
| updateTimer_CPU0_0 | output | optional | Internal timer update |
| PPI16_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI17_CPU1_0 | input | optional | Private peripheral interrupt |

| | | | |
|---|---|---|---|
| PPI18_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI19_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI20_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI21_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI22_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI23_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI24_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI25_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI26_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI27_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI28_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI29_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI30_CPU1_0 | input | optional | Private peripheral interrupt |
| PPI31_CPU1_0 | input | optional | Private peripheral interrupt |
| CNTVIRQ_CPU1_0 | output | optional | EL1 Virtual timer event (active high) |
| CNTPNSIRQ_CPU1_0 | output | optional | EL1 Physical timer event (active high) |
| CNTHPSIRQ_CPU1_0 | output | optional | Secure EL2 Physical timer event (active high) |
| IRQOUT_CPU1_0 | output | optional | IRQ wakeup |
| FIQOUT_CPU1_0 | output | optional | FIQ wakeup |
| RVBARADDRx_CPU1_0 | input | optional | Configure AArch64 Reset Vector Base Address at reset |
| CFGEND_CPU1_0 | input | optional | Configure exception endianness (SCTLR.EE) |
| reset_CPU1_0 | input | optional | Processor reset, active high |
| fiq_CPU1_0 | input | optional | FIQ interrupt, active high (negation of nFIQ) |
| irq_CPU1_0 | input | optional | IRQ interrupt, active high (negation of nIRQ) |
| sei_CPU1_0 | input | optional | System error interrupt, active on rising edge (negation of nSEI) |
| vfiq_CPU1_0 | input | optional | Virtual FIQ interrupt, active high (negation of nVFIQ) |
| virq_CPU1_0 | input | optional | Virtual IRQ interrupt, active high (negation of nVIRQ) |
| vsei_CPU1_0 | input | optional | Virtual system error interrupt, active on rising edge (negation of nVSEI) |
| haltReason_CPU1_0 | output | optional | Indicates why core is halted |
| AXI_SLVERR_CPU1_0 | input | optional | AXI external abort type (DECERR=0, SLVERR=1) |
| CP15SDISABLE_CPU1_0 | input | optional | CP15SDISABLE (active high) |
| PMUIRQ_CPU1_0 | output | optional | Performance monitor event (active high) |
| syncCluster_CPU1_0 | output | optional | Cluster synchronization required |
| updateTimer_CPU1_0 | output | optional | Internal timer update |
| PPI16_CPU2_0 | input | optional | Private peripheral interrupt |

| | | | |
|---|---|---|---|
| PPI17_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI18_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI19_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI20_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI21_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI22_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI23_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI24_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI25_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI26_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI27_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI28_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI29_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI30_CPU2_0 | input | optional | Private peripheral interrupt |
| PPI31_CPU2_0 | input | optional | Private peripheral interrupt |
| CNTVIRQ_CPU2_0 | output | optional | EL1 Virtual timer event (active high) |
| CNTPNSIRQ_CPU2_0 | output | optional | EL1 Physical timer event (active high) |
| CNTHPSIRQ_CPU2_0 | output | optional | Secure EL2 Physical timer event (active high) |
| IRQOUT_CPU2_0 | output | optional | IRQ wakeup |
| FIQOUT_CPU2_0 | output | optional | FIQ wakeup |
| RVBARADDRx_CPU2_0 | input | optional | Configure AArch64 Reset Vector Base Address at reset |
| CFGEND_CPU2_0 | input | optional | Configure exception endianness (SCTLR.EE) |
| reset_CPU2_0 | input | optional | Processor reset, active high |
| fiq_CPU2_0 | input | optional | FIQ interrupt, active high (negation of nFIQ) |
| irq_CPU2_0 | input | optional | IRQ interrupt, active high (negation of nIRQ) |
| sei_CPU2_0 | input | optional | System error interrupt, active on rising edge (negation of nSEI) |
| vfiq_CPU2_0 | input | optional | Virtual FIQ interrupt, active high (negation of nVFIQ) |
| virq_CPU2_0 | input | optional | Virtual IRQ interrupt, active high (negation of nVIRQ) |
| vsei_CPU2_0 | input | optional | Virtual system error interrupt, active on rising edge (negation of nVSEI) |
| haltReason_CPU2_0 | output | optional | Indicates why core is halted |
| AXI_SLVERR_CPU2_0 | input | optional | AXI external abort type (DECERR=0, SLVERR=1) |
| CP15SDISABLE_CPU2_0 | input | optional | CP15SDISABLE (active high) |
| PMUIRQ_CPU2_0 | output | optional | Performance monitor event (active high) |
| syncCluster_CPU2_0 | output | optional | Cluster synchronization required |
| updateTimer_CPU2_0 | output | optional | Internal timer update |

| PPI16_CPU3_0 | input | optional | Private peripheral interrupt |
|---|---|---|---|
| PPI17_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI18_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI19_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI20_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI21_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI22_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI23_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI24_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI25_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI26_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI27_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI28_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI29_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI30_CPU3_0 | input | optional | Private peripheral interrupt |
| PPI31_CPU3_0 | input | optional | Private peripheral interrupt |
| CNTVIRQ_CPU3_0 | output | optional | EL1 Virtual timer event (active high) |
| CNTPNSIRQ_CPU3_0 | output | optional | EL1 Physical timer event (active high) |
| CNTHPSIRQ_CPU3_0 | output | optional | Secure EL2 Physical timer event (active high) |
| IRQOUT_CPU3_0 | output | optional | IRQ wakeup |
| FIQOUT_CPU3_0 | output | optional | FIQ wakeup |
| RVBARADDRx_CPU3_0 | input | optional | Configure AArch64 Reset Vector Base Address at reset |
| CFGEND_CPU3_0 | input | optional | Configure exception endianness (SCTLR.EE) |
| reset_CPU3_0 | input | optional | Processor reset, active high |
| fiq_CPU3_0 | input | optional | FIQ interrupt, active high (negation of nFIQ) |
| irq_CPU3_0 | input | optional | IRQ interrupt, active high (negation of nIRQ) |
| sei_CPU3_0 | input | optional | System error interrupt, active on rising edge (negation of nSEI) |
| vfiq_CPU3_0 | input | optional | Virtual FIQ interrupt, active high (negation of nVFIQ) |
| virq_CPU3_0 | input | optional | Virtual IRQ interrupt, active high (negation of nVIRQ) |
| vsei_CPU3_0 | input | optional | Virtual system error interrupt, active on rising edge (negation of nVSEI) |
| haltReason_CPU3_0 | output | optional | Indicates why core is halted |
| AXI_SLVERR_CPU3_0 | input | optional | AXI external abort type (DECERR=0, SLVERR=1) |
| CP15SDISABLE_CPU3_0 | input | optional | CP15SDISABLE (active high) |
| PMUIRQ_CPU3_0 | output | optional | Performance monitor event (active high) |
| syncCluster_CPU3_0 | output | optional | Cluster synchronization required |

| | | | |
|---|---|---|---|
| updateTimer_CPU3_0 | output | optional | Internal timer update |
| PPI16_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI17_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI18_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI19_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI20_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI21_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI22_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI23_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI24_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI25_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI26_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI27_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI28_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI29_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI30_CPU4_0 | input | optional | Private peripheral interrupt |
| PPI31_CPU4_0 | input | optional | Private peripheral interrupt |
| CNTVIRQ_CPU4_0 | output | optional | EL1 Virtual timer event (active high) |
| CNTPNSIRQ_CPU4_0 | output | optional | EL1 Physical timer event (active high) |
| CNTHPSIRQ_CPU4_0 | output | optional | Secure EL2 Physical timer event (active high) |
| IRQOUT_CPU4_0 | output | optional | IRQ wakeup |
| FIQOUT_CPU4_0 | output | optional | FIQ wakeup |
| RVBARADDRx_CPU4_0 | input | optional | Configure AArch64 Reset Vector Base Address at reset |
| CFGEND_CPU4_0 | input | optional | Configure exception endianness (SCTLR.EE) |
| reset_CPU4_0 | input | optional | Processor reset, active high |
| fiq_CPU4_0 | input | optional | FIQ interrupt, active high (negation of nFIQ) |
| irq_CPU4_0 | input | optional | IRQ interrupt, active high (negation of nIRQ) |
| sei_CPU4_0 | input | optional | System error interrupt, active on rising edge (negation of nSEI) |
| vfiq_CPU4_0 | input | optional | Virtual FIQ interrupt, active high (negation of nVFIQ) |
| virq_CPU4_0 | input | optional | Virtual IRQ interrupt, active high (negation of nVIRQ) |
| vsei_CPU4_0 | input | optional | Virtual system error interrupt, active on rising edge (negation of nVSEI) |
| haltReason_CPU4_0 | output | optional | Indicates why core is halted |
| AXI_SLVERR_CPU4_0 | input | optional | AXI external abort type (DECERR=0, SLVERR=1) |
| CP15SDISABLE_CPU4_0 | input | optional | CP15SDISABLE (active high) |
| PMUIRQ_CPU4_0 | output | optional | Performance monitor event (active high) |

| syncCluster_CPU4_0 | output | optional | Cluster synchronization required |
|---|---|---|---|
| updateTimer_CPU4_0 | output | optional | Internal timer update |

Table 6.1: Net Ports

# Chapter 7

# FIFO Ports

This model has no FIFO ports.

# Chapter 8

# Formal Parameters

| Name | Type | Description |
|------|------|-------------|
| verbose | Boolean | Specify verbosity of output |
| suppressCPSWarnings | Boolean | Suppress duplicate warnings generated using ARM_CP_CPSI or ARM_CP_CPSD message identifiers |
| showHiddenRegs | Boolean | Show hidden registers during register tracing |
| UAL | Boolean | Disassemble using UAL syntax |
| disableGICModel | Boolean | Disable the internal GIC model entirely |
| enableGICv3 | Boolean | Enable/disable GICv3 support |
| enableGICv2_64kB_Page | Boolean | Enable 64kB page size for GICv2 memory-mapped register groups (Xilinx Zynq Ultrascale support) |
| supportSTATUSR | Boolean | Enable/disable support for GICv3 GIC[CDV]_STATUSR registers |
| enableVFPAtReset | Boolean | Enable vector floating point (SIMD and VFP) instructions at reset. (Enables cp10/11 in CPACR and sets FPEXC.EN) |
| SVEImplementedSizes | Uns32 | For processors with ARMv8.2 SVE extension, mask of configurable vector sizes (vector length N is configurable if mask contains $1<<((N/128)-1)$) |
| SVEFaultUnknown | Uns64 | For processors with ARMv8.2 SVE extension, UNKNOWN value returned for suppressed or inactive FFR elements |
| enableSystemBus | Boolean | Add 32-bit artifact System bus port, allowing system registers to be externally implemented |
| enableSystemMonitorBus | Boolean | Add 32-bit artifact SystemMonitor bus port, allowing system register accesses to be externally monitored |
| distinctMTCores | Boolean | For multi-threaded (MT) processors, simulate threads as separate cores (otherwise, simulate MT threads as a single entity) |
| compatibility | Enumeration | Specify compatibility mode |
| | ISA | |
| | gdb | |
| | nopSVC | |
| unpredictableR15 | Enumeration | Specify behavior for UNPREDICTABLE uses of AArch32 R15 register |
| | undefined | |
| | nop | |
| | raz_wi | |
| | execute | |
| | assert | |

| | | |
|---|---|---|
| unpredictableModal | Enumeration | Specify behavior for UNPREDICTABLE instructions in certain AArch32 modes (for example, MRS using SPSR in System mode) |
| | undefined | |
| | nop | |
| | assert | |
| maxSIMDUnroll | Uns32 | If SIMD operations are supported, specify the maximum number of parallel SIMD operations to unroll (unrolled operations can be faster, but produce more verbose JIT code) |
| override_debugMask | Uns32 | Specifies debug mask, enabling debug output for model components |
| thumbNoCond | Boolean | Specify whether trapped Thumb instructions set CV=1 and COND field in syndrome (if False, both are zero) |
| enableHostAtomics | Boolean | Enable use of host atomic instructions to emulate load-/store exclusive operations (not architecturally accurate, accellerates parallel simulation) |
| override_numCPUs | Uns32 | Specify the number of cores in a multiprocessor (maximum of 8 for GICv1/GICv2) |
| override_affinityMask | Uns32 | Specify bitmask of implemented affinity bits in format Aff3:Aff2:Aff1:Aff0 (each a byte) |
| override_MPIDR_MT | Boolean | Specifies that processor is multithreaded |
| override_MPIDR_Aff0 | Uns32 | Override Aff0 field in MPIDR/MPIDR_EL1 register |
| override_MPIDR_Aff1 | Uns32 | Override Aff1 field in MPIDR/MPIDR_EL1 register (also possible by writing CLUSTERIDAFF1 configuration net) |
| override_MPIDR_Aff2 | Uns32 | Override Aff2 field in MPIDR/MPIDR_EL1 register (also possible by writing CLUSTERIDAFF2 configuration net) |
| override_MPIDR_Aff3 | Uns32 | Override Aff3 field in MPIDR_EL1 register (also possible by writing CLUSTERIDAFF3 configuration net) |
| override_fcsePresent | Boolean | Specifies that FCSE is present (if true) |
| override_fpexcDexPresent | Boolean | Specifies that the FPEXC.DEX register field is implemented (if true) |
| override_advSIMDPresent | Boolean | Specifies that Advanced SIMD extensions are present (if true) |
| override_vfpPresent | Boolean | Specifies that VFP extensions are present (if true) |
| override_physicalBits | Uns32 | Specifies the implemented physical bus bits (defaults to connected physical bus width) |
| override_timerScaleFactor | Uns32 | Specifies the fraction of MIPS rate to use for MPCore timers (generic timers or global/local/watchdogs depending on implementation). Defaults to 20 for generic timers, 2 for others |
| override_GICD_NSACRPresent | Boolean | Specifies that optional GICD_NSACR distributor registers are present (GICv2 only) |
| override_GICD_PPISRPresent | Boolean | Specifies that implementation-specific GICD_PPISR distributor register is present (GICv1 ICDPPIS/ICPPISR, GICv1 and GICv2 only) |
| override_GICD_SPISRPresent | Boolean | Specifies that implementation-specific GICD_SPISR distributor registers are present (GICv1 ICDSPIS/ICSPISR) |
| override_GICv3_DistributorBase | Uns64 | Specify distributor register block base address (GICv3 only) |
| override_GICv3_E1NWFPresent | Boolean | Specifies that GICR_CTLR.E1NWF is implemented (GICv3 only) |
| override_GIC_PPIMask | Uns32 | Specify bitmask of implemented PPIs in the GIC (e.g. ID16 is 0x0001, ID31 is 0x8000) |
| override_GICCDISABLE | Boolean | Specify initial value of GICCDISABLE |
| override_SCTLR_V | Boolean | Override SCTLR.V with the passed value (enables high vectors; also configurable using VINITHI pin) |

Copyright (c) 2024 Imperas Software Limited      www.ovpworld.org

OVP License. Release 20240902.0      Page 23 of 56

| override_SCTLR_EE | Boolean | Override SCTLR.EE with the passed value (configures exception data endianness; also configurable using CFGEE pin) |
|---|---|---|
| override_SCTLR_TE | Boolean | Override SCTLR.TE with the passed value (configures Thumb state for exception handling; also configurable using TEINIT pin) |
| override_SCTLR_NMFI | Boolean | Override SCTLR.NMFI with the passed value (configures NMFI state for exception handling; also configurable using CFGNMFI pin) |
| override_SCTLR_CP15BEN_Present | Boolean | Enable ARMv7 SCTLR.CP15BEN bit (CP15 barrier enable) |
| override_MIDR | Uns32 | Override MIDR/MIDR_EL1 register |
| override_CTR | Uns32 | Override CTR/CTR_EL0 register |
| override_MPUIR | Uns32 | Override MPUIR register |
| override_CLIDR | Uns32 | Override CLIDR/CLIDR_EL1 register |
| override_AIDR | Uns32 | Override AIDR/AIDR_EL1 register |
| override_CBAR | Uns32 | Override Configuration Base Address Register (corresponds to value on PERIPHBASE configuration input) |
| override_PFR0 | Uns32 | Override ID_PFR0/ID_PFR0_EL1 register |
| override_PFR1 | Uns32 | Override ID_PFR1/ID_PFR1_EL1 register |
| override_PFR2 | Uns32 | Override ID_PFR2/ID_PFR2_EL1 register |
| override_DFR0 | Uns32 | Override ID_DFR0/ID_DFR0_EL1 register |
| override_DFR1 | Uns32 | Override ID_DFR1/ID_DFR1_EL1 register |
| override_AFR0 | Uns32 | Override ID_AFR0/ID_AFR0_EL1 register |
| override_MMFR0 | Uns32 | Override ID_MMFR0/ID_MMFR0_EL1 register |
| override_MMFR1 | Uns32 | Override ID_MMFR1/ID_MMFR1_EL1 register |
| override_MMFR2 | Uns32 | Override ID_MMFR2/ID_MMFR2_EL1 register |
| override_MMFR3 | Uns32 | Override ID_MMFR3/ID_MMFR3_EL1 register |
| override_MMFR4 | Uns32 | Override ID_MMFR4/ID_MMFR4_EL1 register |
| override_MMFR5 | Uns32 | Override ID_MMFR5/ID_MMFR5_EL1 register |
| override_ISAR0 | Uns32 | Override ID_ISAR0/ID_ISAR0_EL1 register |
| override_ISAR1 | Uns32 | Override ID_ISAR1/ID_ISAR1_EL1 register |
| override_ISAR2 | Uns32 | Override ID_ISAR2/ID_ISAR2_EL1 register |
| override_ISAR3 | Uns32 | Override ID_ISAR3/ID_ISAR3_EL1 register |
| override_ISAR4 | Uns32 | Override ID_ISAR4/ID_ISAR4_EL1 register |
| override_ISAR5 | Uns32 | Override ID_ISAR5/ID_ISAR5_EL1 register |
| override_ISAR6 | Uns32 | Override ID_ISAR6/ID_ISAR6_EL1 register |
| override_PMCR | Uns32 | Override PMCR/PMCR_EL0 register (not functionally significant in the model) |
| override_PMCEID0 | Uns64 | Override PMCEID0/PMCEID0_EL0 register (not functionally significant in the model) |
| override_PMCEID1 | Uns64 | Override PMCEID1/PMCEID1_EL0 register (not functionally significant in the model) |
| override_PMMIR | Uns32 | Override PMMIR/PMMIR_EL1 register (not functionally significant in the model) |
| override_DBGDIDR | Uns32 | Override DBGDIDR register (not functionally significant in the model) |
| override_DBGDEVID | Uns32 | Override DBGDEVID register (not functionally significant in the model) |
| override_DBGDEVID1 | Uns32 | Override DBGDEVID1 register (not functionally significant in the model) |
| override_DBGDEVID2 | Uns32 | Override DBGDEVID2 register (not functionally significant in the model) |
| override_FPSID | Uns32 | Override SIMD/VFP FPSID register |
| override_MVFR0 | Uns32 | Override SIMD/VFP MVFR0/MVFR0_EL1 register |
| override_MVFR1 | Uns32 | Override SIMD/VFP MVFR1/MVFR1_EL1 register |

| override_MVFR2 | Uns32 | Override SIMD/VFP MVFR2/MVFR2_EL1 register |
| override_FPEXC | Uns32 | Override SIMD/VFP FPEXC/FPEXC32_EL2 register |
| override_GICC_IIDR | Uns32 | Override GICC_IIDR register (GICv1 ICCIIDR) |
| override_GICD_TYPER | Uns32 | Override GICD_TYPER register (GICv1 ICDICTR) |
| override_GICD_TYPER_ITLines | Uns32 | Override ITLinesNumber field of GICD_TYPER register (GICv1 ICDICTR) |
| override_GICD_TYPER_ESPI | Boolean | Override ESPI field of GICD_TYPER register (GICv3.1 and later) |
| override_GICD_TYPER_ESPI_range | Uns32 | Override ESPI_range field of GICD_TYPER register (GICv3.1 and later) |
| override_GICD_ICFGRN | Uns32 | Override reset value of GICD_ICFGR2...GICD_ICFGRn (GICv1 ICDICFR2...ICDICFRn) |
| override_GICD_IIDR | Uns32 | Override GICD_IIDR register (GICv1 ICDIIDR) |
| override_GICH_VTR | Uns32 | Override GICH_VTR register |
| override_GICR_IIDR | Uns32 | Override GICR_IIDR register (GICv3 and later) |
| override_GICR_TYPER | Uns64 | Override GICR_TYPER register (GICv3 and later) |
| override_GICR_TYPER_PPInum | Uns32 | Override PPInum field of GICR_TYPER register (GICv3.1 and later) |
| override_GITS_IIDR | Uns32 | Override GITS_IIDR register (GICv3 and later) |
| override_GITS_TYPER | Uns64 | Override GITS_TYPER register (GICv3 and later) |
| override_ICCPMRBits | Uns32 | Specify the number of writable bits in GICC_PMR (GICv1 ICCPMR) |
| override_minICCBPR | Uns32 | Specify the minimum possible value for GICC_BPR (GICv1 ICCBPR) |
| override_ERG | Uns32 | Specifies exclusive reservation granule |
| override_CCSIDR_1I | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 1 instruction) |
| override_CCSIDR_1D | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 1 data) |
| override_CCSIDR_2I | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 2 instruction) |
| override_CCSIDR_2D | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 2 data) |
| override_CCSIDR_3I | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 3 instruction) |
| override_CCSIDR_3D | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 3 data) |
| override_CCSIDR_4I | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 4 instruction) |
| override_CCSIDR_4D | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 4 data) |
| override_CCSIDR_5I | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 5 instruction) |
| override_CCSIDR_5D | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 5 data) |
| override_CCSIDR_6I | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 6 instruction) |
| override_CCSIDR_6D | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 6 data) |
| override_CCSIDR_7I | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 7 instruction) |
| override_CCSIDR_7D | Uns64 | Override CCSIDR/CCSIDR_EL1 (level 7 data) |
| override_RMR | Uns32 | Override RMR register alias at highest-implemented exception level |
| override_RVBAR | Uns64 | Override RVBAR register alias at highest-implemented exception level |
| override_MPUIR_EL1 | Uns32 | Override MPUIR_EL1 register |
| override_MPUIR_EL2 | Uns32 | Override MPUIR_EL2 register |
| override_AA64PFR0_EL1 | Uns64 | Override ID_AA64PFR0_EL1 register |
| override_AA64PFR1_EL1 | Uns64 | Override ID_AA64PFR1_EL1 register |
| override_AA64DFR0_EL1 | Uns64 | Override ID_AA64DFR0_EL1 register |
| override_AA64DFR1_EL1 | Uns64 | Override ID_AA64DFR1_EL1 register |
| override_AA64AFR0_EL1 | Uns64 | Override ID_AA64AFR0_EL1 register |
| override_AA64AFR1_EL1 | Uns64 | Override ID_AA64AFR1_EL1 register |
| override_AA64ISAR0_EL1 | Uns64 | Override ID_AA64ISAR0_EL1 register |
| override_AA64ISAR1_EL1 | Uns64 | Override ID_AA64ISAR1_EL1 register |
| override_AA64ISAR2_EL1 | Uns64 | Override ID_AA64ISAR2_EL1 register |
| override_AA64MMFR0_EL1 | Uns64 | Override ID_AA64MMFR0_EL1 register |
| override_AA64MMFR1_EL1 | Uns64 | Override ID_AA64MMFR1_EL1 register |

| | | |
|---|---|---|
| override_AA64MMFR2_EL1 | Uns64 | Override ID_AA64MMFR2_EL1 register |
| override_DCZID_EL0 | Uns32 | Override DCZID_EL0 register |
| override_LORID_EL1 | Uns32 | Override LORID_EL1 register (ARMv8.1 only) |
| override_STRoffsetPC12 | Boolean | Specifies that STR/STR of PC should do so with 12:byte offset from the current instruction (if true), otherwise an 8:byte offset is used |
| override_ignoreBadCp15 | Boolean | Specifies whether invalid coprocessor 15 access should be ignored (if true) or cause Invalid Instruction exceptions (if false) |
| override_SGIDisable | Boolean | Override whether GIC SGIs may be disabled (if true) or are permanently enabled (if false) |
| override_condUndefined | Boolean | Force undefined instructions to take Undefined Instruction exception even if they are conditional |
| override_deviceStrongAligned | Boolean | Force accesses to Device and Strongly Ordered regions to be aligned |
| override_mask_ACTLR_EL1 | Uns64 | Override mask of writable bits in AArch64 ACTLR_EL1 register, or AArch32 non-secure ACTLR/ACTLR2 pair, if implemented |
| override_mask_ACTLR_EL2 | Uns64 | Override mask of writable bits in AArch64 ACTLR_EL2 register, or AArch32 HACTLR/HACTLR2 pair, if implemented |
| override_mask_ACTLR_EL3 | Uns64 | Override mask of writable bits in AArch64 ACTLR_EL3 register, or AArch32 secure ACTLR/ACTLR2 pair, if implemented |
| override_Control_V | Boolean | Override SCTLR.V with the passed value (deprecated, use override_SCTLR_V) |
| override_MainId | Uns32 | Override MIDR register (deprecated, use override_MIDR) |
| override_CacheType | Uns32 | Override CTR register (deprecated, use override_CTR) |
| override_InstructionAttributes0 | Uns32 | Override ID_ISAR0 register (deprecated, use override_ISAR0) |
| override_InstructionAttributes1 | Uns32 | Override ID_ISAR1 register (deprecated, use override_ISAR1) |
| override_InstructionAttributes2 | Uns32 | Override ID_ISAR2 register (deprecated, use override_ISAR2) |
| override_InstructionAttributes3 | Uns32 | Override ID_ISAR3 register (deprecated, use override_ISAR3) |
| override_InstructionAttributes4 | Uns32 | Override ID_ISAR4 register (deprecated, use override_ISAR4) |
| override_InstructionAttributes5 | Uns32 | Override ID_ISAR5 register (deprecated, use override_ISAR5) |

Table 8.1: Parameters that can be set in: MPCORE

## 8.1 Parameter values and limits

These are the formal parameter limits and actual parameter values

| Name | Min | Max | Default | Actual |
|---|---|---|---|---|
| **(Others)** | | | | |
| variant | | | | Cortex-R82MPx5 |
| verbose | | | t | t |
| suppressCPSWarnings | | | f | f |
| showHiddenRegs | | | f | f |
| UAL | | | t | t |

| | | | | |
|---|---|---|---|---|
| disableGICModel | | | f | f |
| enableGICv3 | | | f | f |
| enableGICv2_64kB_Page | | | f | f |
| supportSTATUSR | | | t | t |
| enableVFPAtReset | | | f | f |
| SVEImplementedSizes | 1 | 0xffff | 15 | 15 |
| SVEFaultUnknown | 0 | 0xffffffffffffffff | 0xdfdfdfdfdfdfdfdf | 0xdfdfdfdfdfdfdfdf |
| enableSystemBus | | | f | f |
| enableSystemMonitorBus | | | f | f |
| distinctMTCores | | | f | f |
| compatibility | | | ISA | ISA |
| unpredictableR15 | | | undefined | undefined |
| unpredictableModal | | | undefined | undefined |
| maxSIMDUnroll | 1 | 16 | 2 | 2 |
| override_debugMask | 0 | 0xffffffff | 0 | 0 |
| thumbNoCond | | | f | f |
| enableHostAtomics | | | f | f |
| endian | | | none | none |
| override_numCPUs | 0 | 32 | 5 | 5 |
| override_affinityMask | 0 | 0xffffffff | 0xffff0700 | 0xffff0700 |
| override_MPIDR_MT | | | f | f |
| override_MPIDR_Aff0 | 0 | 255 | 0 | 0 |
| override_MPIDR_Aff1 | 0 | 255 | 0 | 0 |
| override_MPIDR_Aff2 | 0 | 255 | 0 | 0 |
| override_MPIDR_Aff3 | 0 | 255 | 0 | 0 |
| override_fcsePresent | | | f | f |
| override_fpexcDexPresent | | | t | t |
| override_advSIMDPresent | | | f | f |
| override_vfpPresent | | | f | f |
| override_physicalBits | 32 | 52 | 32 | 32 |
| override_timerScaleFactor | 1 | 0x1ff | 20 | 20 |
| override_GICD_NSACRPresent | | | f | f |
| override_GICD_PPISRPresent | | | t | t |
| override_GICD_SPISRPresent | | | t | t |
| override_GICv3_DistributorBase | 0 | 0xffffffffffffffff | 0x2f000000 | 0x2f000000 |
| override_GICv3_E1NWFPresent | | | f | f |
| override_GIC_PPIMask | 0 | 0xffff | 0 | 0 |
| override_GICCDISABLE | | | f | f |
| override_SCTLR_V | | | f | f |
| override_SCTLR_EE | | | f | f |
| override_SCTLR_TE | | | f | f |
| override_SCTLR_NMFI | | | f | f |
| override_SCTLR_CP15BEN_Present | | | t | t |
| override_MIDR | 0 | 0xffffffff | 0x410fd152 | 0x410fd152 |
| override_CTR | 0 | 0xffffffff | 0x94448004 | 0x94448004 |

| override_MPUIR | 0 | 0xffffffff | 0 | 0 |
|---|---|---|---|---|
| override_CLIDR | 0 | 0xffffffff | 0x82000023 | 0x82000023 |
| override_AIDR | 0 | 0xffffffff | 0 | 0 |
| override_CBAR | 0 | 0xffffffff | 0 | 0 |
| override_PFR0 | 0 | 0xffffffff | 0 | 0 |
| override_PFR1 | 0 | 0xffffffff | 0 | 0 |
| override_PFR2 | 0 | 0xffffffff | 0 | 0 |
| override_DFR0 | 0 | 0xffffffff | 0 | 0 |
| override_DFR1 | 0 | 0xffffffff | 0 | 0 |
| override_AFR0 | 0 | 0xffffffff | 0 | 0 |
| override_MMFR0 | 0 | 0xffffffff | 0 | 0 |
| override_MMFR1 | 0 | 0xffffffff | 0 | 0 |
| override_MMFR2 | 0 | 0xffffffff | 0 | 0 |
| override_MMFR3 | 0 | 0xffffffff | 0 | 0 |
| override_MMFR4 | 0 | 0xffffffff | 0 | 0 |
| override_MMFR5 | 0 | 0xffffffff | 0 | 0 |
| override_ISAR0 | 0 | 0xffffffff | 0 | 0 |
| override_ISAR1 | 0 | 0xffffffff | 0 | 0 |
| override_ISAR2 | 0 | 0xffffffff | 0 | 0 |
| override_ISAR3 | 0 | 0xffffffff | 0 | 0 |
| override_ISAR4 | 0 | 0xffffffff | 0 | 0 |
| override_ISAR5 | 0 | 0xffffffff | 0 | 0 |
| override_ISAR6 | 0 | 0xffffffff | 0 | 0 |
| override_PMCR | 0 | 0xffffffff | 0x414a3000 | 0x414a3000 |
| override_PMCEID0 | 0 | 0xffffffffffffffff | 0xfbff7f3f | 0xfbff7f3f |
| override_PMCEID1 | 0 | 0xffffffffffffffff | 0x1f1ae7b | 0x1f1ae7b |
| override_PMMIR | 0 | 0xffffffff | 0 | 0 |
| override_DBGDIDR | 0 | 0xffffffff | 0x3518c000 | 0x3518c000 |
| override_DBGDEVID | 0 | 0xffffffff | 0x100f10 | 0x100f10 |
| override_DBGDEVID1 | 0 | 0xffffffff | 0 | 0 |
| override_DBGDEVID2 | 0 | 0xffffffff | 0 | 0 |
| override_FPSID | 0 | 0xffffffff | 0x41034080 | 0x41034080 |
| override_MVFR0 | 0 | 0xffffffff | 0x10110222 | 0x10110222 |
| override_MVFR1 | 0 | 0xffffffff | 0x13211111 | 0x13211111 |
| override_MVFR2 | 0 | 0xffffffff | 67 | 67 |
| override_FPEXC | 0 | 0xffffffff | 0 | 0 |
| override_GICC_IIDR | 0 | 0xffffffff | 0x4043b | 0x4043b |
| override_GICD_TYPER | 0 | 0xffffffff | 0x7bfc02 | 0x7bfc02 |
| override_GICD_TYPER_ITLines | 0 | 31 | 2 | 2 |
| override_GICD_TYPER_ESPI | | | f | f |
| override_GICD_TYPER_ESPI_range | 0 | 31 | 0 | 0 |
| override_GICD_ICFGRN | 0 | 0xffffffff | 0 | 0 |
| override_GICD_IIDR | 0 | 0xffffffff | 0x102043b | 0x102043b |
| override_GICH_VTR | 0 | 0xffffffff | 0x90080003 | 0x90080003 |
| override_GICR_IIDR | 0 | 0xffffffff | 0x43b | 0x43b |

| | | | | |
|---|---|---|---|---|
| override_GICR_TYPER | 0 | 0xffffffffffffffff | 9 | 9 |
| override_GICR_TYPER_PPInum | 0 | 2 | 0 | 0 |
| override_GITS_IIDR | 0 | 0xffffffff | 0x43b | 0x43b |
| override_GITS_TYPER | 0 | 0xffffffffffffffff | 0x9ef79 | 0x9ef79 |
| override_ICCPMRBits | 4 | 8 | 5 | 5 |
| override_minICCBPR | 0 | 7 | 2 | 2 |
| override_ERG | 3 | 11 | 4 | 4 |
| override_CCSIDR_1I | 0 | 0xffffffffffffffff | 0x7e01a | 0x7e01a |
| override_CCSIDR_1D | 0 | 0xffffffffffffffff | 0x7e01a | 0x7e01a |
| override_CCSIDR_2I | 0 | 0xffffffffffffffff | 0 | 0 |
| override_CCSIDR_2D | 0 | 0xffffffffffffffff | 0x1fe01a | 0x1fe01a |
| override_CCSIDR_3I | 0 | 0xffffffffffffffff | 0 | 0 |
| override_CCSIDR_3D | 0 | 0xffffffffffffffff | 0 | 0 |
| override_CCSIDR_4I | 0 | 0xffffffffffffffff | 0 | 0 |
| override_CCSIDR_4D | 0 | 0xffffffffffffffff | 0 | 0 |
| override_CCSIDR_5I | 0 | 0xffffffffffffffff | 0 | 0 |
| override_CCSIDR_5D | 0 | 0xffffffffffffffff | 0 | 0 |
| override_CCSIDR_6I | 0 | 0xffffffffffffffff | 0 | 0 |
| override_CCSIDR_6D | 0 | 0xffffffffffffffff | 0 | 0 |
| override_CCSIDR_7I | 0 | 0xffffffffffffffff | 0 | 0 |
| override_CCSIDR_7D | 0 | 0xffffffffffffffff | 0 | 0 |
| override_RMR | 0 | 0xffffffff | 1 | 1 |
| override_RVBAR | 0 | 0xffffffffffffffff | 0 | 0 |
| override_MPUIR_EL1 | 16 | 32 | 16 | 16 |
| override_MPUIR_EL2 | 16 | 32 | 16 | 16 |
| override_AA64PFR0_EL1 | 0 | 0xffffffffffffffff | 0x21110111 | 0x21110111 |
| override_AA64PFR1_EL1 | 0 | 0xffffffffffffffff | 32 | 32 |
| override_AA64DFR0_EL1 | 0 | 0xffffffffffffffff | 0x10305519 | 0x10305519 |
| override_AA64DFR1_EL1 | 0 | 0xffffffffffffffff | 0 | 0 |
| override_AA64AFR0_EL1 | 0 | 0xffffffffffffffff | 0 | 0 |
| override_AA64AFR1_EL1 | 0 | 0xffffffffffffffff | 0 | 0 |
| override_AA64ISAR0_EL1 | 0 | 0xffffffffffffffff | 0x10210000 | 0x10210000 |
| override_AA64ISAR1_EL1 | 0 | 0xffffffffffffffff | 0x5211052 | 0x5211052 |
| override_AA64ISAR2_EL1 | 0 | 0xffffffffffffffff | 0 | 0 |
| override_AA64MMFR0_EL1 | 0 | 0xffffffffffffffff | 0x101122 | 0x101122 |
| override_AA64MMFR1_EL1 | 0 | 0xffffffffffffffff | 0x10200000 | 0x10200000 |
| override_AA64MMFR2_EL1 | 0 | 0xffffffffffffffff | 0x1010 | 0x1010 |
| override_DCZID_EL0 | 0 | 9 | 4 | 4 |
| override_LORID_EL1 | 0 | 0xffffffff | 0x40004 | 0x40004 |
| override_STRoffsetPC12 | | | t | t |
| override_ignoreBadCp15 | | | f | f |
| override_SGIDisable | | | f | f |
| override_condUndefined | | | f | f |
| override_deviceStrongAligned | | | t | t |
| override_mask_ACTLR_EL1 | 0 | 0xffffffffffffffff | 0 | 0 |

Copyright (c) 2024 Imperas Software Limited    www.ovpworld.org

OVP License. Release 20240902.0    Page 29 of 56

| override_mask_ACTLR_EL2 | 0 | 0xffffffffffffffff | 0x3ff | 0x3ff |
|---|---|---|---|---|
| override_mask_ACTLR_EL3 | 0 | 0xffffffffffffffff | 0 | 0 |
| override_Control_V | | | f | f |
| override_MainId | 0 | 0xffffffff | 0x410fd152 | 0x410fd152 |
| override_CacheType | 0 | 0xffffffff | 0x94448004 | 0x94448004 |
| override_InstructionAttributes0 | 0 | 0xffffffff | 0 | 0 |
| override_InstructionAttributes1 | 0 | 0xffffffff | 0 | 0 |
| override_InstructionAttributes2 | 0 | 0xffffffff | 0 | 0 |
| override_InstructionAttributes3 | 0 | 0xffffffff | 0 | 0 |
| override_InstructionAttributes4 | 0 | 0xffffffff | 0 | 0 |
| override_InstructionAttributes5 | 0 | 0xffffffff | 0 | 0 |

Table 8.2: Parameter values and limits

Copyright (c) 2024 Imperas Software Limited     www.ovpworld.org

OVP License. Release 20240902.0     Page 30 of 56

# Chapter 9

# Execution Modes

| Mode | Code |
|------|------|
| EL0t | 0 |
| EL1t | 4 |
| EL1h | 5 |
| EL2t | 8 |
| EL2h | 9 |

Table 9.1: Modes implemented in: CPU

# Chapter 10

# Exceptions

| Exception | Code |
|---|---|
| Reset | 0 |
| Undefined | 1 |
| SupervisorCall | 2 |
| HypervisorCall | 4 |
| PrefetchAbort | 5 |
| DataAbort | 6 |
| HypervisorTrap | 7 |
| IRQ | 8 |
| FIQ | 9 |
| IllegalState | 10 |
| MisalignedPC | 11 |
| MisalignedSP | 12 |
| SError | 13 |

Table 10.1: Exceptions implemented in: CPU

# Chapter 11

# Hierarchy of the model

A CPU core may be configured to instance many processors of a Symmetrical Multi Processor (SMP). A CPU core may also have sub elements within a processor, for example hardware threading blocks.

OVP processor models can be written to include SMP blocks and to have many levels of hierarchy. Some OVP CPU models may have a fixed hierarchy, and some may be configured by settings in a configuration register. Please see the register definitions of this model.

This model documentation shows the settings and hierarchy of the default settings for this model variant.

## 11.1   Level 1: MPCORE

This level in the model hierarchy has 2 commands.
This level in the model hierarchy has no register groups.
This level in the model hierarchy has 5 children:
CPU0, CPU1, CPU2, CPU3 and CPU4.

## 11.2   Level 2: MPCORE

This level in the model hierarchy has 2 commands.
This level in the model hierarchy has no register groups.
This level in the model hierarchy has one child:
CPU0_0

## 11.3   Level 3: CPU

This level in the model hierarchy has 4 commands.
This level in the model hierarchy has 12 register groups:

| Group name | Registers |
|---|---|
| Core_AArch64 | 33 |
| SIMD_FP_AArch64 | 32 |
| AArch64_system | 363 |
| AArch64_system_artifact | 1 |

| AArch64_SYS_instruction_registers | 102 |
|---|---|
| Integration_support | 4 |
| MPCore_distributor | 146 |
| MPCore_physical_redistributor | 43 |
| MPCore_processor_interface | 16 |
| MPCore_virtual_interface_control | 11 |
| MPCore_virtual_processor_interface | 31 |
| MPCore_ITS | 14 |

Table 11.1: Register groups

This level in the model hierarchy has no children.

Copyright (c) 2024 Imperas Software Limited     www.ovpworld.org

OVP License. Release 20240902.0     Page 34 of 56

# Chapter 12

# Model Commands

A Processor model can implement one or more **Model Commands** available to be invoked from the simulator command line, from the OP API or from the Imperas Multiprocessor Debugger.

## 12.1 Level 1: MPCORE

### 12.1.1 isync

specify instruction address range for synchronous execution

| **Argument** | Type | Description |
|---|---|---|
| -addresshi | Uns64 | end address of synchronous execution range |
| -addresslo | Uns64 | start address of synchronous execution range |

Table 12.1: isync command arguments

### 12.1.2 itrace

enable or disable instruction tracing

| **Argument** | Type | Description |
|---|---|---|
| -access | String | show memory accesses by this instruction. Argument can be any combination of X (execute), A (load or store access) and S (system) |
| -after | Uns64 | apply after this many instructions |
| -enable | Boolean | enable instruction tracing |
| -full | Boolean | turn on all trace features |
| -instructioncount | Boolean | include the instruction number in each trace |
| -memory | String | (Alias for access). show memory accesses by this instruction. Argument can be any combination of X (execute), A (load or store access) and S (system) |
| -mode | Boolean | show processor mode changes |
| -off | Boolean | disable instruction tracing |
| -on | Boolean | enable instruction tracing |
| -processorname | Boolean | Include processor name in all trace lines |

| | | |
|---|---|---|
| -registerchange | Boolean | show registers changed by this instruction |
| -registers | Boolean | show registers after each trace |

Table 12.2: itrace command arguments

## 12.2 Level 2: MPCORE

### 12.2.1 isync

specify instruction address range for synchronous execution

| **Argument** | Type | Description |
|---|---|---|
| -addresshi | Uns64 | end address of synchronous execution range |
| -addresslo | Uns64 | start address of synchronous execution range |

Table 12.3: isync command arguments

### 12.2.2 itrace

enable or disable instruction tracing

| **Argument** | Type | Description |
|---|---|---|
| -access | String | show memory accesses by this instruction. Argument can be any combination of X (execute), A (load or store access) and S (system) |
| -after | Uns64 | apply after this many instructions |
| -enable | Boolean | enable instruction tracing |
| -full | Boolean | turn on all trace features |
| -instructioncount | Boolean | include the instruction number in each trace |
| -memory | String | (Alias for access). show memory accesses by this instruction. Argument can be any combination of X (execute), A (load or store access) and S (system) |
| -mode | Boolean | show processor mode changes |
| -off | Boolean | disable instruction tracing |
| -on | Boolean | enable instruction tracing |
| -processorname | Boolean | Include processor name in all trace lines |
| -registerchange | Boolean | show registers changed by this instruction |
| -registers | Boolean | show registers after each trace |

Table 12.4: itrace command arguments

## 12.3 Level 3: CPU

### 12.3.1 debugflags

show or modify the processor debug flags

Copyright (c) 2024 Imperas Software Limited      www.ovpworld.org

OVP License. Release 20240902.0      Page 36 of 56

| Argument | Type | Description |
|----------|------|-------------|
| -get | Boolean | print current processor flags value |
| -mask | Boolean | print valid debug flag bits |
| -set | Int32 | new processor flags (only flags 0x000003e4 can be modified) |

Table 12.5: debugflags command arguments

### 12.3.2    isync

specify instruction address range for synchronous execution

| Argument | Type | Description |
|----------|------|-------------|
| -addresshi | Uns64 | end address of synchronous execution range |
| -addresslo | Uns64 | start address of synchronous execution range |

Table 12.6: isync command arguments

### 12.3.3    itrace

enable or disable instruction tracing

| Argument | Type | Description |
|----------|------|-------------|
| -access | String | show memory accesses by this instruction. Argument can be any combination of X (execute), A (load or store access) and S (system) |
| -after | Uns64 | apply after this many instructions |
| -enable | Boolean | enable instruction tracing |
| -full | Boolean | turn on all trace features |
| -instructioncount | Boolean | include the instruction number in each trace |
| -memory | String | (Alias for access). show memory accesses by this instruction. Argument can be any combination of X (execute), A (load or store access) and S (system) |
| -mode | Boolean | show processor mode changes |
| -off | Boolean | disable instruction tracing |
| -on | Boolean | enable instruction tracing |
| -processorname | Boolean | Include processor name in all trace lines |
| -registerchange | Boolean | show registers changed by this instruction |
| -registers | Boolean | show registers after each trace |

Table 12.7: itrace command arguments

### 12.3.4    listSysRegsAA64

#### 12.3.4.1    Argument description

List all AArch64 system registers

# Chapter 13

# Registers

## 13.1 Level 1: MPCORE

No registers.

## 13.2 Level 2: MPCORE

No registers.

## 13.3 Level 3: CPU

### 13.3.1 Core_AArch64

Registers at level:3, type:CPU group:Core_AArch64

| Name | Bits | Initial-Hex | RW | Description |
|------|------|-------------|-----|-------------|
| x0 | 64 | 0 | rw | |
| x1 | 64 | 0 | rw | |
| x2 | 64 | 0 | rw | |
| x3 | 64 | 0 | rw | |
| x4 | 64 | 0 | rw | |
| x5 | 64 | 0 | rw | |
| x6 | 64 | 0 | rw | |
| x7 | 64 | 0 | rw | |
| x8 | 64 | 0 | rw | |
| x9 | 64 | 0 | rw | |
| x10 | 64 | 0 | rw | |
| x11 | 64 | 0 | rw | |
| x12 | 64 | 0 | rw | |
| x13 | 64 | 0 | rw | |
| x14 | 64 | 0 | rw | |
| x15 | 64 | 0 | rw | |
| x16 | 64 | 0 | rw | |
| x17 | 64 | 0 | rw | |
| x18 | 64 | 0 | rw | |
| x19 | 64 | 0 | rw | |
| x20 | 64 | 0 | rw | |
| x21 | 64 | 0 | rw | |
| x22 | 64 | 0 | rw | |

| x23 | 64 | 0 | rw | |
|-----|-----|---|-----|---|
| x24 | 64 | 0 | rw | |
| x25 | 64 | 0 | rw | |
| x26 | 64 | 0 | rw | |
| x27 | 64 | 0 | rw | |
| x28 | 64 | 0 | rw | |
| x29 | 64 | 0 | rw | frame pointer |
| x30 | 64 | 0 | rw | |
| sp | 64 | 0 | rw | stack pointer |
| pc | 64 | 0 | rw | program counter |

Table 13.1: Registers at level 3, type:CPU group:Core_AArch64

### 13.3.2   SIMD_FP_AArch64

Registers at level:3, type:CPU group:SIMD_FP_AArch64

| Name | Bits | Initial-Hex | RW | Description |
|------|------|-------------|-----|-------------|
| v0 | 128 | - | rw | |
| v1 | 128 | - | rw | |
| v2 | 128 | - | rw | |
| v3 | 128 | - | rw | |
| v4 | 128 | - | rw | |
| v5 | 128 | - | rw | |
| v6 | 128 | - | rw | |
| v7 | 128 | - | rw | |
| v8 | 128 | - | rw | |
| v9 | 128 | - | rw | |
| v10 | 128 | - | rw | |
| v11 | 128 | - | rw | |
| v12 | 128 | - | rw | |
| v13 | 128 | - | rw | |
| v14 | 128 | - | rw | |
| v15 | 128 | - | rw | |
| v16 | 128 | - | rw | |
| v17 | 128 | - | rw | |
| v18 | 128 | - | rw | |
| v19 | 128 | - | rw | |
| v20 | 128 | - | rw | |
| v21 | 128 | - | rw | |
| v22 | 128 | - | rw | |
| v23 | 128 | - | rw | |
| v24 | 128 | - | rw | |
| v25 | 128 | - | rw | |
| v26 | 128 | - | rw | |
| v27 | 128 | - | rw | |
| v28 | 128 | - | rw | |
| v29 | 128 | - | rw | |
| v30 | 128 | - | rw | |
| v31 | 128 | - | rw | |

Table 13.2: Registers at level 3, type:CPU group:SIMD_FP_AArch64

### 13.3.3 AArch64_system

Registers at level:3, type:CPU group:AArch64_system

| Name | Bits | Initial-Hex | RW | Description |
|------|------|-------------|----|-------------|
| ACTLR_EL1 | 64 | 0 | rw | Auxiliary Control (EL1) |
| ACTLR_EL2 | 64 | 0 | rw | Auxiliary Control (EL2) |
| AFSR0_EL1 | 32 | 0 | rw | Auxiliary Fault Status 0 (EL1) |
| AFSR0_EL2 | 32 | 0 | rw | Auxiliary Fault Status 0 (EL2) |
| AFSR1_EL1 | 32 | 0 | rw | Auxiliary Fault Status 1 (EL1) |
| AFSR1_EL2 | 32 | 0 | rw | Auxiliary Fault Status 1 (EL2) |
| AIDR_EL1 | 32 | 0 | r- | Auxiliary ID |
| AMAIR_EL1 | 64 | 0 | rw | Auxiliary Memory Attribute Indirection (EL1) |
| AMAIR_EL2 | 64 | 0 | rw | Auxiliary Memory Attribute Indirection (EL2) |
| APDAKeyHi_EL1 | 64 | 0 | rw | Pointer Authentication Key A for Data (high bits) |
| APDAKeyLo_EL1 | 64 | 0 | rw | Pointer Authentication Key A for Data (low bits) |
| APDBKeyHi_EL1 | 64 | 0 | rw | Pointer Authentication Key B for Data (high bits) |
| APDBKeyLo_EL1 | 64 | 0 | rw | Pointer Authentication Key B for Data (low bits) |
| APGAKeyHi_EL1 | 64 | 0 | rw | Pointer Authentication Key A for Code (high bits) |
| APGAKeyLo_EL1 | 64 | 0 | rw | Pointer Authentication Key A for Code (low bits) |
| APIAKeyHi_EL1 | 64 | 0 | rw | Pointer Authentication Key A for Instruction (high bits) |
| APIAKeyLo_EL1 | 64 | 0 | rw | Pointer Authentication Key A for Instruction (low bits) |
| APIBKeyHi_EL1 | 64 | 0 | rw | Pointer Authentication Key B for Instruction (high bits) |
| APIBKeyLo_EL1 | 64 | 0 | rw | Pointer Authentication Key B for Instruction (low bits) |
| CCSIDR_EL1 | 32 | 7e01a | r- | Current Cache Size ID |
| CLIDR_EL1 | 32 | 82000023 | r- | Cache Level ID |
| CNTFRQ_EL0 | 32 | 4c4b40 | rw | Counter-Timer Frequency |
| CNTHCTL_EL2 | 32 | 3 | rw | Counter-Timer Hypervisor Control |
| CNTHPS_CTL_EL2 | 32 | 0 | rw | Counter-Timer Secure Physical Timer Control (EL2) |
| CNTHPS_CVAL_EL2 | 64 | 0 | rw | Counter-Timer Secure Physical Timer CompareValue (EL2) |
| CNTHPS_TVAL_EL2 | 32 | 0 | rw | Counter-Timer Secure Physical Timer TimerValue (EL2) |
| CNTKCTL_EL1 | 32 | 0 | rw | Counter-Timer Kernel Control |
| CNTPCT_EL0 | 64 | 0 | r- | Counter-Timer Physical Count |
| CNTP_CTL_EL0 | 32 | 0 | rw | Counter-Timer Physical Timer Control |
| CNTP_CVAL_EL0 | 64 | 0 | rw | Counter-Timer Physical Timer CompareValue |
| CNTP_TVAL_EL0 | 32 | 0 | rw | Counter-Timer Physical Timer TimerValue |
| CNTVCT_EL0 | 64 | 0 | r- | Counter-Timer Virtual Count |
| CNTVOFF_EL2 | 64 | 0 | rw | Counter-Timer Virtual Offset |
| CNTV_CTL_EL0 | 32 | 0 | rw | Counter-Timer Virtual Timer Control |

| CNTV_CVAL_EL0 | 64 | 0 | rw | Counter-Timer Virtual Timer CompareValue |
|---|---|---|---|---|
| CNTV_TVAL_EL0 | 32 | 0 | rw | Counter-Timer Virtual Timer Timer-Value |
| CONTEXTIDR_EL1 | 32 | 0 | rw | Context ID (EL1) |
| CONTEXTIDR_EL2 | 32 | 0 | rw | Context ID (EL2) |
| CPACR_EL1 | 32 | 0 | rw | Architectural Feature Access Control |
| CPTR_EL2 | 32 | 33ff | rw | Architectural Feature Trap (EL2) |
| CSSELR_EL1 | 32 | 0 | rw | Current Size Selection |
| CTR_EL0 | 32 | 94448004 | r- | Cache Type |
| CurrentEL | 32 | 8 | r- | Current Exception Level |
| DAIF | 32 | 3c0 | rw | Interrupt Mask Bits |
| DBGAUTHSTATUS_EL1 | 32 | 0 | r- | Debug Authentication Status |
| DBGBCR0_EL1 | 32 | 1e0 | rw | Debug Breakpoint Control 0 |
| DBGBCR1_EL1 | 32 | 1e0 | rw | Debug Breakpoint Control 1 |
| DBGBCR2_EL1 | 32 | 1e0 | rw | Debug Breakpoint Control 2 |
| DBGBCR3_EL1 | 32 | 1e0 | rw | Debug Breakpoint Control 3 |
| DBGBCR4_EL1 | 32 | 1e0 | rw | Debug Breakpoint Control 4 |
| DBGBCR5_EL1 | 32 | 1e0 | rw | Debug Breakpoint Control 5 |
| DBGBVR0_EL1 | 64 | 0 | rw | Debug Breakpoint Value 0 |
| DBGBVR1_EL1 | 64 | 0 | rw | Debug Breakpoint Value 1 |
| DBGBVR2_EL1 | 64 | 0 | rw | Debug Breakpoint Value 2 |
| DBGBVR3_EL1 | 64 | 0 | rw | Debug Breakpoint Value 3 |
| DBGBVR4_EL1 | 64 | 0 | rw | Debug Breakpoint Value 4 |
| DBGBVR5_EL1 | 64 | 0 | rw | Debug Breakpoint Value 5 |
| DBGCLAIMCLR_EL1 | 32 | 0 | rw | Debug Claim Tag Clear |
| DBGCLAIMSET_EL1 | 32 | 0 | rw | Debug Claim Tag Set |
| DBGDTRTRX_EL0 | 32 | 0 | rw | Debug Data Transfer, Transmit/Receive |
| DBGDTR_EL0 | 64 | 0 | rw | Debug Data Transfer |
| DBGPRCR_EL1 | 32 | 0 | rw | Debug Power Control |
| DBGWCR0_EL1 | 32 | 0 | rw | Debug Watchpoint Control 0 |
| DBGWCR1_EL1 | 32 | 0 | rw | Debug Watchpoint Control 1 |
| DBGWCR2_EL1 | 32 | 0 | rw | Debug Watchpoint Control 2 |
| DBGWCR3_EL1 | 32 | 0 | rw | Debug Watchpoint Control 3 |
| DBGWVR0_EL1 | 64 | 0 | rw | Debug Watchpoint Value 0 |
| DBGWVR1_EL1 | 64 | 0 | rw | Debug Watchpoint Value 1 |
| DBGWVR2_EL1 | 64 | 0 | rw | Debug Watchpoint Value 2 |
| DBGWVR3_EL1 | 64 | 0 | rw | Debug Watchpoint Value 3 |
| DCZID_EL0 | 32 | 4 | r- | Data Cache Zero ID |
| DISR_EL1 | 32 | 0 | rw | Deferred Interrupt Status |
| DIT | 32 | 0 | rw | Data Independent Timing |
| DLR_EL0 | 64 | 0 | rw | Debug Link |
| DSPSR_EL0 | 32 | 0 | rw | Debug Saved Program Status |
| ELR_EL1 | 64 | 0 | rw | Exception Link (EL1) |
| ELR_EL2 | 64 | 0 | rw | Exception Link (EL2) |
| ERRIDR_EL1 | 32 | 0 | r- | Error Record ID |
| ERRSELR_EL1 | 32 | 0 | rw | Error Record Select |
| ERXADDR_EL1 | 64 | 0 | rw | Selected Error Record Address |
| ERXCTLR_EL1 | 64 | 0 | rw | Selected Error Record Control |
| ERXFR_EL1 | 64 | 0 | r- | Selected Error Record Feature |
| ERXMISC0_EL1 | 64 | 0 | rw | Selected Error Record Miscellaneous 0 |
| ERXMISC1_EL1 | 64 | 0 | rw | Selected Error Record Miscellaneous 1 |

| ERXMISC2_EL1 | 64 | 0 | rw | Selected Error Record Miscellaneous 2 |
|---|---|---|---|---|
| ERXMISC3_EL1 | 64 | 0 | rw | Selected Error Record Miscellaneous 3 |
| ERXPFGCDN_EL1 | 32 | 0 | rw | Selected Pseudo-fault Generation Countdown |
| ERXPFGCTL_EL1 | 32 | 0 | rw | Selected Pseudo-fault Generation Control |
| ERXPFGF_EL1 | 32 | 0 | r- | Selected Pseudo-fault Generation Feature |
| ERXSTATUS_EL1 | 32 | 0 | rw | Selected Error Record Status |
| ESR_EL1 | 32 | 0 | rw | Exception Syndrome (EL1) |
| ESR_EL2 | 32 | 0 | rw | Exception Syndrome (EL2) |
| FAR_EL1 | 64 | 0 | rw | Fault Address (EL1) |
| FAR_EL2 | 64 | 0 | rw | Fault Address (EL2) |
| FPCR | 32 | 0 | rw | Floating Point Control |
| FPSR | 32 | 0 | rw | Floating Point Status |
| HACR_EL2 | 32 | 0 | rw | Hypervisor Auxiliary Control |
| HCR_EL2 | 64 | 80000002 | rw | Hypervisor Configuration |
| HPFAR_EL2 | 64 | 0 | rw | Hypervisor IPA Fault Address |
| HSTR_EL2 | 32 | 0 | rw | Hypervisor System Trap |
| ICC_AP0R0_EL1 | 32 | 0 | rw | Interrupt Controller Active Priorities Group 0, 0 |
| ICC_AP1R0_EL1 | 32 | 0 | rw | Interrupt Controller Active Priorities Group 1, 0 |
| ICC_ASGI1R_EL1 | 64 | - | -w | Interrupt Controller Alias SGI Group 1 |
| ICC_BPR0_EL1 | 32 | 2 | rw | Interrupt Controller Binary Point 0 |
| ICC_BPR1_EL1 | 32 | 2 | rw | Interrupt Controller Binary Point 1 |
| ICC_CTLR_EL1 | 32 | 80400 | rw | Interrupt Controller Control (EL1) |
| ICC_DIR_EL1 | 32 | - | -w | Interrupt Controller Deactivate Interrupt |
| ICC_EOIR0_EL1 | 32 | - | -w | Interrupt Controller End of Interrupt 0 |
| ICC_EOIR1_EL1 | 32 | - | -w | Interrupt Controller End of Interrupt 1 |
| ICC_HPPIR0_EL1 | 32 | 3ff | r- | Interrupt Controller Highest Priority Pending Interrupt 0 |
| ICC_HPPIR1_EL1 | 32 | 3ff | r- | Interrupt Controller Highest Priority Pending Interrupt 1 |
| ICC_IAR0_EL1 | 32 | 3ff | r- | Interrupt Controller Interrupt Acknowledge 0 |
| ICC_IAR1_EL1 | 32 | 3ff | r- | Interrupt Controller Interrupt Acknowledge 1 |
| ICC_IGRPEN0_EL1 | 32 | 0 | rw | Interrupt Controller Interrupt Group 0 Enable |
| ICC_IGRPEN1_EL1 | 32 | 0 | rw | Interrupt Controller Interrupt Group 1 Enable |
| ICC_PMR_EL1 | 32 | 0 | rw | Interrupt Controller Priority Mask |
| ICC_RPR_EL1 | 32 | ff | r- | Interrupt Controller Running Priority |
| ICC_SGI0R_EL1 | 64 | - | -w | Interrupt Controller SGI Group 0 |
| ICC_SGI1R_EL1 | 64 | - | -w | Interrupt Controller SGI Group 1 |
| ICC_SRE_EL1 | 32 | 1 | rw | Interrupt Controller System Register Enable (EL1) |
| ICC_SRE_EL2 | 32 | 9 | rw | Interrupt Controller System Register Enable (EL2) |

| Register | Size | Reset | Access | Description |
|---|---|---|---|---|
| ICH_AP0R0_EL2 | 32 | 0 | rw | Interrupt Controller Hyp Active Priorities Group 0 (Word 0) |
| ICH_AP1R0_EL2 | 32 | 0 | rw | Interrupt Controller Hyp Active Priorities Group 1 (Word 0) |
| ICH_EISR_EL2 | 32 | 0 | r- | Interrupt Controller End of Interrupt Status |
| ICH_ELRSR_EL2 | 32 | f | r- | Interrupt Controller Empty List Register Status |
| ICH_HCR_EL2 | 32 | 0 | rw | Interrupt Controller Hypervisor Control |
| ICH_LR0_EL2 | 64 | 0 | rw | Interrupt Controller List 0 |
| ICH_LR1_EL2 | 64 | 0 | rw | Interrupt Controller List 1 |
| ICH_LR2_EL2 | 64 | 0 | rw | Interrupt Controller List 2 |
| ICH_LR3_EL2 | 64 | 0 | rw | Interrupt Controller List 3 |
| ICH_MISR_EL2 | 32 | 0 | r- | Interrupt Controller Maintenance Interrupt State |
| ICH_VMCR_EL2 | 32 | 4c0000 | rw | Interrupt Controller Virtual Machine Control |
| ICH_VTR_EL2 | 32 | 90180003 | r- | Interrupt Controller VGIC Type |
| ID_AA64AFR0_EL1 | 64 | 0 | r- | AArch64 Auxiliary Feature 0 |
| ID_AA64AFR1_EL1 | 64 | 0 | r- | AArch64 Auxiliary Feature 1 |
| ID_AA64DFR0_EL1 | 64 | 1f0 10305519 | r- | AArch64 Debug Feature 0 |
| ID_AA64DFR1_EL1 | 64 | 0 | r- | AArch64 Debug Feature 1 |
| ID_AA64ISAR0_EL1 | 64 | 2111000 10210000 | r- | AArch64 Instruction Set Attribute 0 |
| ID_AA64ISAR1_EL1 | 64 | 10110 05211052 | r- | AArch64 Instruction Set Attribute 1 |
| ID_AA64MMFR0_EL1 | 64 | 1f0000 00101122 | r- | AArch64 Memory Model Feature 0 |
| ID_AA64MMFR1_EL1 | 64 | 10200000 | r- | AArch64 Memory Model Feature 1 |
| ID_AA64MMFR2_EL1 | 64 | 111 00001010 | r- | AArch64 Memory Model Feature 2 |
| ID_AA64PFR0_EL1 | 64 | 11010010 21110111 | r- | AArch64 Processor Feature 0 |
| ID_AA64PFR1_EL1 | 64 | 1 00000020 | r- | AArch64 Processor Feature 1 |
| ID_AFR0_EL1 | 32 | 0 | r- | Auxiliary Feature 0 |
| ID_DFR0_EL1 | 32 | 0 | r- | Debug Feature 0 |
| ID_ISAR0_EL1 | 32 | 0 | r- | Instruction Set Attribute 0 |
| ID_ISAR1_EL1 | 32 | 0 | r- | Instruction Set Attribute 1 |
| ID_ISAR2_EL1 | 32 | 0 | r- | Instruction Set Attribute 2 |
| ID_ISAR3_EL1 | 32 | 0 | r- | Instruction Set Attribute 3 |
| ID_ISAR4_EL1 | 32 | 0 | r- | Instruction Set Attribute 4 |
| ID_ISAR5_EL1 | 32 | 0 | r- | Instruction Set Attribute 5 |
| ID_ISAR6_EL1 | 32 | 0 | r- | Instruction Set Attribute 6 |
| ID_MMFR0_EL1 | 32 | 0 | r- | Memory Model Feature 0 |
| ID_MMFR1_EL1 | 32 | 0 | r- | Memory Model Feature 1 |
| ID_MMFR2_EL1 | 32 | 0 | r- | Memory Model Feature 2 |
| ID_MMFR3_EL1 | 32 | 0 | r- | Memory Model Feature 3 |
| ID_MMFR4_EL1 | 32 | 0 | r- | Memory Model Feature 4 |
| ID_PFR0_EL1 | 32 | 0 | r- | Processor Feature 0 |
| ID_PFR1_EL1 | 32 | 0 | r- | Processor Feature 1 |
| ID_PFR2_EL1 | 32 | 0 | r- | Processor Feature 2 |
| IMP_BPCTLR_EL1 | 64 | 222 | rw | Branch Predictor Control |
| IMP_CDBGDR0_EL1 | 64 | 0 | r- | Cache Debug Data 0 |
| IMP_CDBGDR1_EL1 | 64 | 0 | r- | Cache Debug Data 1 |
| IMP_CLUSTERACELSCTLR_EL1 | 64 | 3ffff | rw | ACELS Port Control |

| | | | | |
|---|---|---|---|---|
| IMP_CLUSTERACPSID_EL1 | 32 | 0 | rw | Cluster ACP Scheme ID |
| IMP_CLUSTERACTLR_EL1 | 32 | 0 | rw | Cluster Auxiliary Control |
| IMP_CLUSTERCDBGDR0_EL1 | 64 | 0 | r- | Cache Debug Data |
| IMP_CLUSTERCFR_EL1 | 32 | 5 | r- | Cluster Configuration |
| IMP_CLUSTERMEMPROTCTLR_EL1 | 32 | 0 | rw | Cluster Memory Protection Control |
| IMP_CLUSTERPARTCR_EL1 | 32 | 0 | rw | Cluster Partition Control |
| IMP_CLUSTERPMCCNTR_EL1 | 64 | 0 | rw | Cluster Performance Monitors Cycle Count |
| IMP_CLUSTERPMCEID0_EL1 | 32 | 66020000 | r- | Cluster Common Event ID 0 |
| IMP_CLUSTERPMCEID1_EL1 | 32 | 1e00 | r- | Cluster Common Event ID 1 |
| IMP_CLUSTERPMCLAIMCLR_EL1 | 32 | 0 | rw | Cluster Performance Monitor Claim Tag Clear |
| IMP_CLUSTERPMCLAIMSET_EL1 | 32 | 0 | rw | Cluster Performance Monitor Claim Tag Set |
| IMP_CLUSTERPMCNTENCLR_EL1 | 32 | 0 | rw | Cluster Count Enable Clear |
| IMP_CLUSTERPMCNTENSET_EL1 | 32 | 0 | rw | Cluster Count Enable Set |
| IMP_CLUSTERPMCR_EL1 | 32 | 41413040 | rw | Cluster Performance Monitors Control |
| IMP_CLUSTERPMINTENCLR_EL1 | 32 | 0 | rw | Cluster Interrupt Enable Clear |
| IMP_CLUSTERPMINTENSET_EL1 | 32 | 0 | rw | Cluster Interrupt Enable Set |
| IMP_CLUSTERPMOVSCLR_EL1 | 32 | 0 | rw | Cluster Overflow Flag Status Clear |
| IMP_CLUSTERPMOVSSET_EL1 | 32 | 0 | rw | Cluster Overflow Flag Status Set |
| IMP_CLUSTERPMSELR_EL1 | 32 | 0 | rw | Cluster Event Counter Selection |
| IMP_CLUSTERPMXEVCNTR_EL1 | 32 | 0 | rw | Cluster Selected Event Count |
| IMP_CLUSTERPMXEVTYPER_EL1 | 32 | 0 | rw | Cluster Selected Event Type |
| IMP_CLUSTERPWRCTLR_EL1 | 32 | 0 | rw | Cluster Power Control |
| IMP_CLUSTERPWRDN_EL1 | 32 | 0 | rw | Cluster Powerdown |
| IMP_CLUSTERPWRSTAT_EL1 | 32 | 30 | r- | Cluster Power Status |
| IMP_CLUSTERQOSR_EL1 | 32 | e | rw | Cluster Quality of Service |
| IMP_CLUSTERSID_EL1 | 32 | 0 | rw | Cluster Scheme ID |
| IMP_CLUSTERTESTR1_EL1 | 32 | 0 | rw | Cluster Test 1 |
| IMP_CPUACTLR_EL1 | 64 | 114 e50f387f | rw | CPU Auxiliary Control |
| IMP_CPUCFR_EL1 | 32 | 0 | r- | CPU Configuration |
| IMP_CPUPCR_EL1 | 64 | 0 | rw | Selected Instruction Patch Control |
| IMP_CPUPMR_EL1 | 64 | 0 | rw | Selected Instruction Patch Mask |
| IMP_CPUPOR_EL1 | 64 | 0 | rw | Selected Instruction Patch Opcode |
| IMP_CPUPSELR_EL1 | 32 | 0 | rw | Instruction Patch Selection |
| IMP_CPUPWRCTLR_EL1 | 32 | 0 | rw | CPU Power Control |
| IMP_DTCMREGIONR_EL1 | 64 | 0 | rw | DTCM Region |
| IMP_ITCMREGIONR_EL1 | 64 | 0 | rw | ITCM Region |
| IMP_LATENCY_EL2 | 64 | 0 | rw | Interrupt Latency |
| IMP_LLPPREGIONR_EL1 | 64 | 0 | rw | LLPP Region |
| IMP_LLRAMREGIONR_EL1 | 64 | 0 | rw | LLRAM Region |
| IMP_MEMPROTCTLR_EL1 | 32 | 0 | rw | Memory Protection Control |
| IMP_SPPREGIONR_EL1 | 64 | 0 | rw | SPP Region |
| IMP_TESTR0_EL1 | 64 | 0 | r- | Test 0 |
| IMP_TESTR1_EL1 | 64 | 0 | r- | Test 1 |
| ISR_EL1 | 32 | 0 | r- | Interrupt Status |
| MAIR_EL1 | 64 | 0 | rw | Memory Attribute Indirection (EL1) |
| MAIR_EL2 | 64 | 0 | rw | Memory Attribute Indirection (EL2) |
| MDCCINT_EL1 | 32 | 0 | rw | Monitor DCC Interrupt Enable |
| MDCCSR_EL0 | 32 | 0 | r- | Monitor DCC Status |
| MDCR_EL2 | 32 | 6 | rw | Monitor Debug Configuration (EL2) |
| MDRAR_EL1 | 64 | 0 | r- | Monitor Debug ROM Address |
| MDSCR_EL1 | 32 | 0 | rw | Monitor Debug System Control |
| MIDR_EL1 | 32 | 410fd152 | r- | Main ID |

| MPIDR_EL1 | 64 | 80000000 | r- | Multiprocessor Affinity |
|---|---|---|---|---|
| MPUIR_EL1 | 32 | 10 | r- | MPU Type (EL1) |
| MPUIR_EL2 | 32 | 10 | r- | MPU Type (EL2) |
| MVFR0_EL1 | 32 | 0 | r- | Media and VFP Feature 0 |
| MVFR1_EL1 | 32 | 0 | r- | Media and VFP Feature 1 |
| MVFR2_EL1 | 32 | 0 | r- | Media and VFP Feature 2 |
| NZCV | 32 | 0 | rw | Condition Flags |
| OSDLR_EL1 | 32 | 0 | rw | OS Double Lock |
| OSDTRRX_EL1 | 32 | 0 | rw | OS Lock Data Transfer, Receive |
| OSDTRTX_EL1 | 32 | 0 | rw | OS Lock Data Transfer, Transmit |
| OSECCR_EL1 | 32 | 0 | rw | OS Lock Exception Catch Control |
| OSLAR_EL1 | 32 | - | -w | OS Lock Access |
| OSLSR_EL1 | 32 | a | r- | OS Lock Status |
| PAN | 32 | 0 | rw | Privileged Access Never |
| PAR_EL1 | 64 | 0 | rw | Physical Address |
| PMCCFILTR_EL0 | 32 | 0 | rw | Performance Monitors Cycle Count Filter |
| PMCCNTR_EL0 | 64 | 0 | rw | Performance Monitors Cycle Count |
| PMCEID0_EL0 | 64 | fbff7f3f | r- | Performance Monitors Common Event ID 0 |
| PMCEID1_EL0 | 64 | 1f1ae7b | r- | Performance Monitors Common Event ID 1 |
| PMCNTENCLR_EL0 | 32 | 0 | rw | Performance Monitors Count Enable Clear |
| PMCNTENSET_EL0 | 32 | 0 | rw | Performance Monitors Count Enable Set |
| PMCR_EL0 | 32 | 414a3040 | rw | Performance Monitors Control |
| PMEVCNTR0_EL0 | 32 | 0 | rw | Performance Monitors Event Count 0 |
| PMEVCNTR1_EL0 | 32 | 0 | rw | Performance Monitors Event Count 1 |
| PMEVCNTR2_EL0 | 32 | 0 | rw | Performance Monitors Event Count 2 |
| PMEVCNTR3_EL0 | 32 | 0 | rw | Performance Monitors Event Count 3 |
| PMEVCNTR4_EL0 | 32 | 0 | rw | Performance Monitors Event Count 4 |
| PMEVCNTR5_EL0 | 32 | 0 | rw | Performance Monitors Event Count 5 |
| PMEVTYPER0_EL0 | 32 | 0 | rw | Performance Monitors Event Type 0 |
| PMEVTYPER1_EL0 | 32 | 0 | rw | Performance Monitors Event Type 1 |
| PMEVTYPER2_EL0 | 32 | 0 | rw | Performance Monitors Event Type 2 |
| PMEVTYPER3_EL0 | 32 | 0 | rw | Performance Monitors Event Type 3 |
| PMEVTYPER4_EL0 | 32 | 0 | rw | Performance Monitors Event Type 4 |
| PMEVTYPER5_EL0 | 32 | 0 | rw | Performance Monitors Event Type 5 |
| PMINTENCLR_EL1 | 32 | 0 | rw | Performance Monitors Interrupt Enable Clear |
| PMINTENSET_EL1 | 32 | 0 | rw | Performance Monitors Interrupt Enable Set |
| PMMIR_EL1 | 32 | 0 | r- | Performance Monitors Machine Identification |
| PMOVSCLR_EL0 | 32 | 0 | rw | Performance Monitors Overflow Flag Status Clear |
| PMOVSSET_EL0 | 32 | 0 | rw | Performance Monitors Overflow Flag Status Set |
| PMSELR_EL0 | 32 | 0 | rw | Performance Monitors Event Counter Selection |
| PMSWINC_EL0 | 32 | - | -w | Performance Monitors Software Increment |
| PMUSERENR_EL0 | 32 | 0 | rw | Performance Monitors User Enable |
| PMXEVCNTR_EL0 | 32 | 0 | rw | Performance Monitors Selected Event Count |

Copyright (c) 2024 Imperas Software Limited     www.ovpworld.org

OVP License. Release 20240902.0     Page 45 of 56

| PMXEVTYPER_EL0 | 32 | 0 | rw | Performance Monitors Selected Event Type |
|---|---|---|---|---|
| PRBAR1_EL1 | 64 | 0 | rw | Protection Region Base Address 1 (EL1) |
| PRBAR1_EL2 | 64 | 0 | rw | Protection Region Base Address 1 (EL2) |
| PRBAR2_EL1 | 64 | 0 | rw | Protection Region Base Address 2 (EL1) |
| PRBAR2_EL2 | 64 | 0 | rw | Protection Region Base Address 2 (EL2) |
| PRBAR3_EL1 | 64 | 0 | rw | Protection Region Base Address 3 (EL1) |
| PRBAR3_EL2 | 64 | 0 | rw | Protection Region Base Address 3 (EL2) |
| PRBAR4_EL1 | 64 | 0 | rw | Protection Region Base Address 4 (EL1) |
| PRBAR4_EL2 | 64 | 0 | rw | Protection Region Base Address 4 (EL2) |
| PRBAR5_EL1 | 64 | 0 | rw | Protection Region Base Address 5 (EL1) |
| PRBAR5_EL2 | 64 | 0 | rw | Protection Region Base Address 5 (EL2) |
| PRBAR6_EL1 | 64 | 0 | rw | Protection Region Base Address 6 (EL1) |
| PRBAR6_EL2 | 64 | 0 | rw | Protection Region Base Address 6 (EL2) |
| PRBAR7_EL1 | 64 | 0 | rw | Protection Region Base Address 7 (EL1) |
| PRBAR7_EL2 | 64 | 0 | rw | Protection Region Base Address 7 (EL2) |
| PRBAR8_EL1 | 64 | 0 | rw | Protection Region Base Address 8 (EL1) |
| PRBAR8_EL2 | 64 | 0 | rw | Protection Region Base Address 8 (EL2) |
| PRBAR9_EL1 | 64 | 0 | rw | Protection Region Base Address 9 (EL1) |
| PRBAR9_EL2 | 64 | 0 | rw | Protection Region Base Address 9 (EL2) |
| PRBAR10_EL1 | 64 | 0 | rw | Protection Region Base Address 10 (EL1) |
| PRBAR10_EL2 | 64 | 0 | rw | Protection Region Base Address 10 (EL2) |
| PRBAR11_EL1 | 64 | 0 | rw | Protection Region Base Address 11 (EL1) |
| PRBAR11_EL2 | 64 | 0 | rw | Protection Region Base Address 11 (EL2) |
| PRBAR12_EL1 | 64 | 0 | rw | Protection Region Base Address 12 (EL1) |
| PRBAR12_EL2 | 64 | 0 | rw | Protection Region Base Address 12 (EL2) |
| PRBAR13_EL1 | 64 | 0 | rw | Protection Region Base Address 13 (EL1) |
| PRBAR13_EL2 | 64 | 0 | rw | Protection Region Base Address 13 (EL2) |
| PRBAR14_EL1 | 64 | 0 | rw | Protection Region Base Address 14 (EL1) |

| PRBAR14_EL2 | 64 | 0 | rw | Protection Region Base Address 14 (EL2) |
|---|---|---|---|---|
| PRBAR15_EL1 | 64 | 0 | rw | Protection Region Base Address 15 (EL1) |
| PRBAR15_EL2 | 64 | 0 | rw | Protection Region Base Address 15 (EL2) |
| PRBAR_EL1 | 64 | 0 | rw | Protection Region Base Address (EL1) |
| PRBAR_EL2 | 64 | 0 | rw | Protection Region Base Address (EL2) |
| PRENR_EL1 | 32 | 0 | rw | Protection Region Enable (EL1) |
| PRENR_EL2 | 32 | 0 | rw | Protection Region Enable (EL2) |
| PRLAR1_EL1 | 64 | 0 | rw | Protection Region Limit Address 1 (EL1) |
| PRLAR1_EL2 | 64 | 0 | rw | Protection Region Limit Address 1 (EL2) |
| PRLAR2_EL1 | 64 | 0 | rw | Protection Region Limit Address 2 (EL1) |
| PRLAR2_EL2 | 64 | 0 | rw | Protection Region Limit Address 2 (EL2) |
| PRLAR3_EL1 | 64 | 0 | rw | Protection Region Limit Address 3 (EL1) |
| PRLAR3_EL2 | 64 | 0 | rw | Protection Region Limit Address 3 (EL2) |
| PRLAR4_EL1 | 64 | 0 | rw | Protection Region Limit Address 4 (EL1) |
| PRLAR4_EL2 | 64 | 0 | rw | Protection Region Limit Address 4 (EL2) |
| PRLAR5_EL1 | 64 | 0 | rw | Protection Region Limit Address 5 (EL1) |
| PRLAR5_EL2 | 64 | 0 | rw | Protection Region Limit Address 5 (EL2) |
| PRLAR6_EL1 | 64 | 0 | rw | Protection Region Limit Address 6 (EL1) |
| PRLAR6_EL2 | 64 | 0 | rw | Protection Region Limit Address 6 (EL2) |
| PRLAR7_EL1 | 64 | 0 | rw | Protection Region Limit Address 7 (EL1) |
| PRLAR7_EL2 | 64 | 0 | rw | Protection Region Limit Address 7 (EL2) |
| PRLAR8_EL1 | 64 | 0 | rw | Protection Region Limit Address 8 (EL1) |
| PRLAR8_EL2 | 64 | 0 | rw | Protection Region Limit Address 8 (EL2) |
| PRLAR9_EL1 | 64 | 0 | rw | Protection Region Limit Address 9 (EL1) |
| PRLAR9_EL2 | 64 | 0 | rw | Protection Region Limit Address 9 (EL2) |
| PRLAR10_EL1 | 64 | 0 | rw | Protection Region Limit Address 10 (EL1) |
| PRLAR10_EL2 | 64 | 0 | rw | Protection Region Limit Address 10 (EL2) |
| PRLAR11_EL1 | 64 | 0 | rw | Protection Region Limit Address 11 (EL1) |
| PRLAR11_EL2 | 64 | 0 | rw | Protection Region Limit Address 11 (EL2) |

| PRLAR12_EL1 | 64 | 0 | rw | Protection Region Limit Address 12 (EL1) |
|---|---|---|---|---|
| PRLAR12_EL2 | 64 | 0 | rw | Protection Region Limit Address 12 (EL2) |
| PRLAR13_EL1 | 64 | 0 | rw | Protection Region Limit Address 13 (EL1) |
| PRLAR13_EL2 | 64 | 0 | rw | Protection Region Limit Address 13 (EL2) |
| PRLAR14_EL1 | 64 | 0 | rw | Protection Region Limit Address 14 (EL1) |
| PRLAR14_EL2 | 64 | 0 | rw | Protection Region Limit Address 14 (EL2) |
| PRLAR15_EL1 | 64 | 0 | rw | Protection Region Limit Address 15 (EL1) |
| PRLAR15_EL2 | 64 | 0 | rw | Protection Region Limit Address 15 (EL2) |
| PRLAR_EL1 | 64 | 0 | rw | Protection Region Limit Address (EL1) |
| PRLAR_EL2 | 64 | 0 | rw | Protection Region Limit Address (EL2) |
| PRSELR_EL1 | 32 | 0 | rw | Protection Region Selection (EL1) |
| PRSELR_EL2 | 32 | 0 | rw | Protection Region Selection (EL2) |
| REVIDR_EL1 | 32 | 0 | r- | Revision ID |
| RVBAR_EL2 | 64 | 0 | r- | Reset Vector Base Address (EL2) |
| SCTLR_EL1 | 64 | 30d50980 | rw | System Control (EL1) |
| SCTLR_EL2 | 64 | 30c50830 | rw | System Control (EL2) |
| SPSR_EL1 | 32 | 0 | rw | Saved Program Status (EL1) |
| SPSR_EL2 | 32 | 0 | rw | Saved Program Status (EL2) |
| SPSR_abt | 32 | 0 | rw | Saved Program Status (Abort Mode) |
| SPSR_fiq | 32 | 0 | rw | Saved Program Status (FIQ Mode) |
| SPSR_irq | 32 | 0 | rw | Saved Program Status (IRQ Mode) |
| SPSR_und | 32 | 0 | rw | Saved Program Status (Undefined Mode) |
| SPSel | 32 | 1 | rw | Stack Pointer Select |
| SP_EL0 | 64 | 0 | rw | Stack Pointer (EL0) |
| SP_EL1 | 64 | 0 | rw | Stack Pointer (EL1) |
| SP_EL2 | 64 | 0 | rw | Stack Pointer (EL2) |
| SSBS | 32 | 0 | rw | Speculative Store Bypass |
| TCR_EL1 | 64 | 0 | rw | Translation Control (EL1) |
| TCR_EL2 | 32 | 80800000 | rw | Translation Control (EL2) |
| TPIDRRO_EL0 | 64 | 0 | rw | Thread Pointer/ID, Read-Only (EL0) |
| TPIDR_EL0 | 64 | 0 | rw | Thread Pointer/ID (EL0) |
| TPIDR_EL1 | 64 | 0 | rw | Thread Pointer/ID (EL1) |
| TPIDR_EL2 | 64 | 0 | rw | Thread Pointer/ID (EL2) |
| TRFCR_EL1 | 32 | 0 | rw | Trace Filter Control (EL1) |
| TRFCR_EL2 | 32 | 0 | rw | Trace Filter Control (EL2) |
| TTBR0_EL1 | 64 | 0 | rw | Translation Table Base 0 (EL1) |
| UAO | 32 | 0 | rw | Unprivileged Access Override |
| VBAR_EL1 | 64 | 0 | rw | Vector Base Address (EL1) |
| VBAR_EL2 | 64 | 0 | rw | Vector Base Address (EL2) |
| VDISR_EL2 | 32 | 0 | rw | Virtual Deferred Interrupt Status |
| VMPIDR_EL2 | 64 | 80000000 | rw | Virtualization Multiprocessor ID |
| VPIDR_EL2 | 32 | 410fd152 | rw | Virtualization Processor ID |
| VSCTLR_EL2 | 64 | 0 | rw | Virtualization System Control |
| VSESR_EL2 | 32 | 0 | rw | Virtual SError Exception Syndrome |

Copyright (c) 2024 Imperas Software Limited     www.ovpworld.org

OVP License. Release 20240902.0     Page 48 of 56

| VSTCR_EL2 | 64 | 80000000 | rw | Virtualization Secure Translation Control |
| VTCR_EL2 | 32 | 0 | rw | Virtualization Translation Control |

Table 13.3: Registers at level 3, type:CPU group:AArch64_system

### 13.3.4 AArch64_system_artifact

Registers at level:3, type:CPU group:AArch64_system_artifact

| **Name** | Bits | Initial-Hex | RW | Description |
| --- | --- | --- | --- | --- |
| cpsr | 32 | 3c9 | rw | |

Table 13.4: Registers at level 3, type:CPU group:AArch64_system_artifact

### 13.3.5 AArch64_SYS_instruction_registers

Registers at level:3, type:CPU group:AArch64_SYS_instruction_registers

| **Name** | Bits | Initial-Hex | RW | Description |
| --- | --- | --- | --- | --- |
| ATS1E0R | 64 | - | -w | |
| ATS1E0W | 64 | - | -w | |
| ATS1E1R | 64 | - | -w | |
| ATS1E1RP | 64 | - | -w | |
| ATS1E1W | 64 | - | -w | |
| ATS1E1WP | 64 | - | -w | |
| ATS1E2R | 64 | - | -w | |
| ATS1E2W | 64 | - | -w | |
| ATS12E0R | 64 | - | -w | |
| ATS12E0W | 64 | - | -w | |
| ATS12E1R | 64 | - | -w | |
| ATS12E1W | 64 | - | -w | |
| CFPRCTX | 64 | - | -w | |
| CPPRCTX | 64 | - | -w | |
| DCCISW | 32 | - | -w | |
| DCCIVAC | 64 | - | -w | |
| DCCSW | 32 | - | -w | |
| DCCVAC | 64 | - | -w | |
| DCCVADP | 64 | - | -w | |
| DCCVAP | 64 | - | -w | |
| DCCVAU | 64 | - | -w | |
| DCISW | 32 | - | -w | |
| DCIVAC | 64 | - | -w | |
| DCZVA | 32 | - | -w | |
| DVPRCTX | 64 | - | -w | |
| ICIALLU | 32 | - | -w | |
| ICIALLUIS | 32 | - | -w | |
| ICIVAU | 64 | - | -w | |
| SYS_IMP_BPI | 64 | - | -w | |
| SYS_IMP_CDBGDCD | 64 | - | -w | |
| SYS_IMP_CDBGDCT | 64 | - | -w | |
| SYS_IMP_CDBGICD | 64 | - | -w | |
| SYS_IMP_CDBGICT | 64 | - | -w | |
| SYS_IMP_CDBGTD | 64 | - | -w | |
| SYS_IMP_CDBGTT | 64 | - | -w | |

| SYS_IMP_CLUSTERCDBGL2D | 64 | - | -w | |
|---|---|---|---|---|
| SYS_IMP_CLUSTERCDBGL2DT | 64 | - | -w | |
| SYS_IMP_CLUSTERCDBGL2T | 64 | - | -w | |
| SYS_IMP_CLUSTERCDBGLCUDT | 64 | - | -w | |
| TLBIALLE1 | 64 | - | -w | |
| TLBIALLE1IS | 64 | - | -w | |
| TLBIALLE1OS | 64 | - | -w | |
| TLBIALLE2 | 64 | - | -w | |
| TLBIALLE2IS | 64 | - | -w | |
| TLBIALLE2OS | 64 | - | -w | |
| TLBIASIDE1 | 64 | - | -w | |
| TLBIASIDE1IS | 64 | - | -w | |
| TLBIASIDE1OS | 64 | - | -w | |
| TLBIIPAS2E1 | 64 | - | -w | |
| TLBIIPAS2E1IS | 64 | - | -w | |
| TLBIIPAS2E1OS | 64 | - | -w | |
| TLBIIPAS2LE1 | 64 | - | -w | |
| TLBIIPAS2LE1IS | 64 | - | -w | |
| TLBIIPAS2LE1OS | 64 | - | -w | |
| TLBIRIPAS2E1 | 64 | - | -w | |
| TLBIRIPAS2E1IS | 64 | - | -w | |
| TLBIRIPAS2E1OS | 64 | - | -w | |
| TLBIRIPAS2LE1 | 64 | - | -w | |
| TLBIRIPAS2LE1IS | 64 | - | -w | |
| TLBIRIPAS2LE1OS | 64 | - | -w | |
| TLBIRVAAE1 | 64 | - | -w | |
| TLBIRVAAE1IS | 64 | - | -w | |
| TLBIRVAAE1OS | 64 | - | -w | |
| TLBIRVAALE1 | 64 | - | -w | |
| TLBIRVAALE1IS | 64 | - | -w | |
| TLBIRVAALE1OS | 64 | - | -w | |
| TLBIRVAE1 | 64 | - | -w | |
| TLBIRVAE1IS | 64 | - | -w | |
| TLBIRVAE1OS | 64 | - | -w | |
| TLBIRVAE2 | 64 | - | -w | |
| TLBIRVAE2IS | 64 | - | -w | |
| TLBIRVAE2OS | 64 | - | -w | |
| TLBIRVALE1 | 64 | - | -w | |
| TLBIRVALE1IS | 64 | - | -w | |
| TLBIRVALE1OS | 64 | - | -w | |
| TLBIRVALE2 | 64 | - | -w | |
| TLBIRVALE2IS | 64 | - | -w | |
| TLBIRVALE2OS | 64 | - | -w | |
| TLBIVAAE1 | 64 | - | -w | |
| TLBIVAAE1IS | 64 | - | -w | |
| TLBIVAAE1OS | 64 | - | -w | |
| TLBIVAALE1 | 64 | - | -w | |
| TLBIVAALE1IS | 64 | - | -w | |
| TLBIVAALE1OS | 64 | - | -w | |
| TLBIVAE1 | 64 | - | -w | |
| TLBIVAE1IS | 64 | - | -w | |
| TLBIVAE1OS | 64 | - | -w | |
| TLBIVAE2 | 64 | - | -w | |
| TLBIVAE2IS | 64 | - | -w | |
| TLBIVAE2OS | 64 | - | -w | |
| TLBIVALE1 | 64 | - | -w | |

| | | | | |
|---|---|---|---|---|
| TLBIVALE1IS | 64 | - | -w | |
| TLBIVALE1OS | 64 | - | -w | |
| TLBIVALE2 | 64 | - | -w | |
| TLBIVALE2IS | 64 | - | -w | |
| TLBIVALE2OS | 64 | - | -w | |
| TLBIVMALLE1 | 64 | - | -w | |
| TLBIVMALLE1IS | 64 | - | -w | |
| TLBIVMALLE1OS | 64 | - | -w | |
| TLBIVMALLS12E1 | 64 | - | -w | |
| TLBIVMALLS12E1IS | 64 | - | -w | |
| TLBIVMALLS12E1OS | 64 | - | -w | |

Table 13.5: Registers at level 3, type:CPU group:AArch64_SYS_instruction_registers

### 13.3.6   Integration_support

Registers at level:3, type:CPU group:Integration_support

| Name | Bits | Initial-Hex | RW | Description |
|---|---|---|---|---|
| transactPL | 32 | 2 | r- | privilege level of current memory transaction |
| transactAT | 32 | 0 | r- | current memory transaction type: PA=1, VA=0 |
| HaltReason | 8 | 0 | r- | bit field indicating halt reason |
| atomicType | 8 | 0 | r- | current atomic instruction type (1:atomic, 2:exclusive) |

Table 13.6: Registers at level 3, type:CPU group:Integration_support

### 13.3.7   MPCore_distributor

Registers at level:3, type:CPU group:MPCore_distributor

| Name | Bits | Initial-Hex | RW | Description |
|---|---|---|---|---|
| GICD_CIDR0 | 32 | d | r- | Component ID 0 |
| GICD_CIDR1 | 32 | f0 | r- | Component ID 1 |
| GICD_CIDR2 | 32 | 5 | r- | Component ID 2 |
| GICD_CIDR3 | 32 | b1 | r- | Component ID 3 |
| GICD_CLRSPI_NSR | 32 | 0 | -w | Clear SPI |
| GICD_CLRSPI_SR | 32 | 0 | -w | Clear SPI, Secure |
| GICD_CTLR | 32 | 30 | rw | Distributor Control |
| GICD_ICACTIVER0 | 32 | 0 | rw | Interrupt Clear-Active 0 |
| GICD_ICACTIVER1 | 32 | 0 | rw | Interrupt Clear-Active 1 |
| GICD_ICACTIVER2 | 32 | 0 | rw | Interrupt Clear-Active 2 |
| GICD_ICENABLER0 | 32 | ffff | rw | Interrupt Clear-Enable 0 |
| GICD_ICENABLER1 | 32 | 0 | rw | Interrupt Clear-Enable 1 |
| GICD_ICENABLER2 | 32 | 0 | rw | Interrupt Clear-Enable 2 |
| GICD_ICFGR0 | 32 | aaaaaaaa | rw | Interrupt Configuration 0 |
| GICD_ICFGR1 | 32 | 0 | rw | Interrupt Configuration 1 |
| GICD_ICFGR2 | 32 | 0 | rw | Interrupt Configuration 2 |
| GICD_ICFGR3 | 32 | 0 | rw | Interrupt Configuration 3 |
| GICD_ICFGR4 | 32 | 0 | rw | Interrupt Configuration 4 |
| GICD_ICFGR5 | 32 | 0 | rw | Interrupt Configuration 5 |
| GICD_ICPENDR0 | 32 | 0 | rw | Interrupt Clear-Pending 0 |
| GICD_ICPENDR1 | 32 | 0 | rw | Interrupt Clear-Pending 1 |
| GICD_ICPENDR2 | 32 | 0 | rw | Interrupt Clear-Pending 2 |
| GICD_IGROUPR0 | 32 | 0 | rw | Interrupt Group 0 |
| GICD_IGROUPR1 | 32 | 0 | rw | Interrupt Group 1 |

Copyright (c) 2024 Imperas Software Limited                              www.ovpworld.org

OVP License. Release 20240902.0                                          Page 51 of 56

| GICD_IGROUPR2 | 32 | 0 | rw | Interrupt Group 2 |
|---|---|---|---|---|
| GICD_IGRPMODR0 | 32 | 0 | rw | Interrupt Group Modifier 0 |
| GICD_IGRPMODR1 | 32 | 0 | rw | Interrupt Group Modifier 1 |
| GICD_IGRPMODR2 | 32 | 0 | rw | Interrupt Group Modifier 2 |
| GICD_IIDR | 32 | 102043b | r- | Distributor Implementor ID |
| GICD_IPRIORITYR0 | 32 | 0 | rw | Interrupt Priority 0 |
| GICD_IPRIORITYR1 | 32 | 0 | rw | Interrupt Priority 1 |
| GICD_IPRIORITYR2 | 32 | 0 | rw | Interrupt Priority 2 |
| GICD_IPRIORITYR3 | 32 | 0 | rw | Interrupt Priority 3 |
| GICD_IPRIORITYR4 | 32 | 0 | rw | Interrupt Priority 4 |
| GICD_IPRIORITYR5 | 32 | 0 | rw | Interrupt Priority 5 |
| GICD_IPRIORITYR6 | 32 | 0 | rw | Interrupt Priority 6 |
| GICD_IPRIORITYR7 | 32 | 0 | rw | Interrupt Priority 7 |
| GICD_IPRIORITYR8 | 32 | 0 | rw | Interrupt Priority 8 |
| GICD_IPRIORITYR9 | 32 | 0 | rw | Interrupt Priority 9 |
| GICD_IPRIORITYR10 | 32 | 0 | rw | Interrupt Priority 10 |
| GICD_IPRIORITYR11 | 32 | 0 | rw | Interrupt Priority 11 |
| GICD_IPRIORITYR12 | 32 | 0 | rw | Interrupt Priority 12 |
| GICD_IPRIORITYR13 | 32 | 0 | rw | Interrupt Priority 13 |
| GICD_IPRIORITYR14 | 32 | 0 | rw | Interrupt Priority 14 |
| GICD_IPRIORITYR15 | 32 | 0 | rw | Interrupt Priority 15 |
| GICD_IPRIORITYR16 | 32 | 0 | rw | Interrupt Priority 16 |
| GICD_IPRIORITYR17 | 32 | 0 | rw | Interrupt Priority 17 |
| GICD_IPRIORITYR18 | 32 | 0 | rw | Interrupt Priority 18 |
| GICD_IPRIORITYR19 | 32 | 0 | rw | Interrupt Priority 19 |
| GICD_IPRIORITYR20 | 32 | 0 | rw | Interrupt Priority 20 |
| GICD_IPRIORITYR21 | 32 | 0 | rw | Interrupt Priority 21 |
| GICD_IPRIORITYR22 | 32 | 0 | rw | Interrupt Priority 22 |
| GICD_IPRIORITYR23 | 32 | 0 | rw | Interrupt Priority 23 |
| GICD_IROUTER32 | 64 | 0 | rw | Interrupt Routing 32 |
| GICD_IROUTER33 | 64 | 0 | rw | Interrupt Routing 33 |
| GICD_IROUTER34 | 64 | 0 | rw | Interrupt Routing 34 |
| GICD_IROUTER35 | 64 | 0 | rw | Interrupt Routing 35 |
| GICD_IROUTER36 | 64 | 0 | rw | Interrupt Routing 36 |
| GICD_IROUTER37 | 64 | 0 | rw | Interrupt Routing 37 |
| GICD_IROUTER38 | 64 | 0 | rw | Interrupt Routing 38 |
| GICD_IROUTER39 | 64 | 0 | rw | Interrupt Routing 39 |
| GICD_IROUTER40 | 64 | 0 | rw | Interrupt Routing 40 |
| GICD_IROUTER41 | 64 | 0 | rw | Interrupt Routing 41 |
| GICD_IROUTER42 | 64 | 0 | rw | Interrupt Routing 42 |
| GICD_IROUTER43 | 64 | 0 | rw | Interrupt Routing 43 |
| GICD_IROUTER44 | 64 | 0 | rw | Interrupt Routing 44 |
| GICD_IROUTER45 | 64 | 0 | rw | Interrupt Routing 45 |
| GICD_IROUTER46 | 64 | 0 | rw | Interrupt Routing 46 |
| GICD_IROUTER47 | 64 | 0 | rw | Interrupt Routing 47 |
| GICD_IROUTER48 | 64 | 0 | rw | Interrupt Routing 48 |
| GICD_IROUTER49 | 64 | 0 | rw | Interrupt Routing 49 |
| GICD_IROUTER50 | 64 | 0 | rw | Interrupt Routing 50 |
| GICD_IROUTER51 | 64 | 0 | rw | Interrupt Routing 51 |
| GICD_IROUTER52 | 64 | 0 | rw | Interrupt Routing 52 |
| GICD_IROUTER53 | 64 | 0 | rw | Interrupt Routing 53 |
| GICD_IROUTER54 | 64 | 0 | rw | Interrupt Routing 54 |
| GICD_IROUTER55 | 64 | 0 | rw | Interrupt Routing 55 |
| GICD_IROUTER56 | 64 | 0 | rw | Interrupt Routing 56 |
| GICD_IROUTER57 | 64 | 0 | rw | Interrupt Routing 57 |
| GICD_IROUTER58 | 64 | 0 | rw | Interrupt Routing 58 |

| GICD_IROUTER59 | 64 | 0 | rw | Interrupt Routing 59 |
|---|---|---|---|---|
| GICD_IROUTER60 | 64 | 0 | rw | Interrupt Routing 60 |
| GICD_IROUTER61 | 64 | 0 | rw | Interrupt Routing 61 |
| GICD_IROUTER62 | 64 | 0 | rw | Interrupt Routing 62 |
| GICD_IROUTER63 | 64 | 0 | rw | Interrupt Routing 63 |
| GICD_IROUTER64 | 64 | 0 | rw | Interrupt Routing 64 |
| GICD_IROUTER65 | 64 | 0 | rw | Interrupt Routing 65 |
| GICD_IROUTER66 | 64 | 0 | rw | Interrupt Routing 66 |
| GICD_IROUTER67 | 64 | 0 | rw | Interrupt Routing 67 |
| GICD_IROUTER68 | 64 | 0 | rw | Interrupt Routing 68 |
| GICD_IROUTER69 | 64 | 0 | rw | Interrupt Routing 69 |
| GICD_IROUTER70 | 64 | 0 | rw | Interrupt Routing 70 |
| GICD_IROUTER71 | 64 | 0 | rw | Interrupt Routing 71 |
| GICD_IROUTER72 | 64 | 0 | rw | Interrupt Routing 72 |
| GICD_IROUTER73 | 64 | 0 | rw | Interrupt Routing 73 |
| GICD_IROUTER74 | 64 | 0 | rw | Interrupt Routing 74 |
| GICD_IROUTER75 | 64 | 0 | rw | Interrupt Routing 75 |
| GICD_IROUTER76 | 64 | 0 | rw | Interrupt Routing 76 |
| GICD_IROUTER77 | 64 | 0 | rw | Interrupt Routing 77 |
| GICD_IROUTER78 | 64 | 0 | rw | Interrupt Routing 78 |
| GICD_IROUTER79 | 64 | 0 | rw | Interrupt Routing 79 |
| GICD_IROUTER80 | 64 | 0 | rw | Interrupt Routing 80 |
| GICD_IROUTER81 | 64 | 0 | rw | Interrupt Routing 81 |
| GICD_IROUTER82 | 64 | 0 | rw | Interrupt Routing 82 |
| GICD_IROUTER83 | 64 | 0 | rw | Interrupt Routing 83 |
| GICD_IROUTER84 | 64 | 0 | rw | Interrupt Routing 84 |
| GICD_IROUTER85 | 64 | 0 | rw | Interrupt Routing 85 |
| GICD_IROUTER86 | 64 | 0 | rw | Interrupt Routing 86 |
| GICD_IROUTER87 | 64 | 0 | rw | Interrupt Routing 87 |
| GICD_IROUTER88 | 64 | 0 | rw | Interrupt Routing 88 |
| GICD_IROUTER89 | 64 | 0 | rw | Interrupt Routing 89 |
| GICD_IROUTER90 | 64 | 0 | rw | Interrupt Routing 90 |
| GICD_IROUTER91 | 64 | 0 | rw | Interrupt Routing 91 |
| GICD_IROUTER92 | 64 | 0 | rw | Interrupt Routing 92 |
| GICD_IROUTER93 | 64 | 0 | rw | Interrupt Routing 93 |
| GICD_IROUTER94 | 64 | 0 | rw | Interrupt Routing 94 |
| GICD_IROUTER95 | 64 | 0 | rw | Interrupt Routing 95 |
| GICD_ISACTIVER0 | 32 | 0 | rw | Interrupt Set-Active 0 |
| GICD_ISACTIVER1 | 32 | 0 | rw | Interrupt Set-Active 1 |
| GICD_ISACTIVER2 | 32 | 0 | rw | Interrupt Set-Active 2 |
| GICD_ISENABLER0 | 32 | ffff | rw | Interrupt Set-Enable 0 |
| GICD_ISENABLER1 | 32 | 0 | rw | Interrupt Set-Enable 1 |
| GICD_ISENABLER2 | 32 | 0 | rw | Interrupt Set-Enable 2 |
| GICD_ISPENDR0 | 32 | 0 | rw | Interrupt Set-Pending 0 |
| GICD_ISPENDR1 | 32 | 0 | rw | Interrupt Set-Pending 1 |
| GICD_ISPENDR2 | 32 | 0 | rw | Interrupt Set-Pending 2 |
| GICD_NSACR0 | 32 | 0 | rw | Interrupt Non-secure Access Control 0 |
| GICD_NSACR1 | 32 | 0 | rw | Interrupt Non-secure Access Control 1 |
| GICD_NSACR2 | 32 | 0 | rw | Interrupt Non-secure Access Control 2 |
| GICD_NSACR3 | 32 | 0 | rw | Interrupt Non-secure Access Control 3 |
| GICD_NSACR4 | 32 | 0 | rw | Interrupt Non-secure Access Control 4 |
| GICD_NSACR5 | 32 | 0 | rw | Interrupt Non-secure Access Control 5 |
| GICD_PIDR0 | 32 | 92 | r- | Peripheral ID 0 |
| GICD_PIDR1 | 32 | b4 | r- | Peripheral ID 1 |
| GICD_PIDR2 | 32 | 3b | r- | Peripheral ID 2 |
| GICD_PIDR3 | 32 | 0 | r- | Peripheral ID 3 |

| GICD_PIDR4 | 32 | 44 | r- | Peripheral ID 4 |
|---|---|---|---|---|
| GICD_PIDR5 | 32 | 0 | r- | Peripheral ID 5 |
| GICD_PIDR6 | 32 | 0 | r- | Peripheral ID 6 |
| GICD_PIDR7 | 32 | 0 | r- | Peripheral ID 7 |
| GICD_SETSPI_NSR | 32 | 0 | -w | Set SPI |
| GICD_SETSPI_SR | 32 | 0 | -w | Set SPI, Secure |
| GICD_SPISR0 | 32 | 0 | r- | SPI Status 0 |
| GICD_SPISR1 | 32 | 0 | r- | SPI Status 1 |
| GICD_STATUSR | 32 | 0 | rw | Distributor Status |
| GICD_TYPER | 32 | 7b0402 | r- | Interrupt Controller Type |

Table 13.7: Registers at level 3, type:CPU group:MPCore_distributor

### 13.3.8    MPCore_physical_redistributor

Registers at level:3, type:CPU group:MPCore_physical_redistributor

| Name | Bits | Initial-Hex | RW | Description |
|---|---|---|---|---|
| GICR_CIDR0 | 32 | d | r- | Redistributor Component ID 0 |
| GICR_CIDR1 | 32 | f0 | r- | Redistributor Component ID 1 |
| GICR_CIDR2 | 32 | 5 | r- | Redistributor Component ID 2 |
| GICR_CIDR3 | 32 | b1 | r- | Redistributor Component ID 3 |
| GICR_CLRLPIR | 64 | 0 | -w | Clear LPI Pending |
| GICR_CTLR | 32 | 0 | rw | Redistributor Control |
| GICR_ICACTIVER0 | 32 | 0 | rw | Interrupt Clear-Active 0 |
| GICR_ICENABLER0 | 32 | ffff | rw | Interrupt Clear-Enable 0 |
| GICR_ICFGR0 | 32 | aaaaaaaa | rw | Interrupt Configuration 0 |
| GICR_ICFGR1 | 32 | 0 | rw | Interrupt Configuration 1 |
| GICR_ICPENDR0 | 32 | 0 | rw | Interrupt Clear-Pending 0 |
| GICR_IGROUPR0 | 32 | 0 | rw | Interrupt Group 0 |
| GICR_IGRPMODR0 | 32 | 0 | rw | Interrupt Group Modifier 0 |
| GICR_IIDR | 32 | 43b | r- | Redistributor Implementer Identification |
| GICR_INVALLR | 64 | 0 | -w | Redistributor Invalidate All |
| GICR_INVLPIR | 64 | 0 | -w | Redistributor Invalidate LPI |
| GICR_IPRIORITYR0 | 32 | 0 | rw | Interrupt Priority 0 |
| GICR_IPRIORITYR1 | 32 | 0 | rw | Interrupt Priority 1 |
| GICR_IPRIORITYR2 | 32 | 0 | rw | Interrupt Priority 2 |
| GICR_IPRIORITYR3 | 32 | 0 | rw | Interrupt Priority 3 |
| GICR_IPRIORITYR4 | 32 | 0 | rw | Interrupt Priority 4 |
| GICR_IPRIORITYR5 | 32 | 0 | rw | Interrupt Priority 5 |
| GICR_IPRIORITYR6 | 32 | 0 | rw | Interrupt Priority 6 |
| GICR_IPRIORITYR7 | 32 | 0 | rw | Interrupt Priority 7 |
| GICR_ISACTIVER0 | 32 | 0 | rw | Interrupt Set-Active 0 |
| GICR_ISENABLER0 | 32 | ffff | rw | Interrupt Set-Enable 0 |
| GICR_ISPENDR0 | 32 | 0 | rw | Interrupt Set-Pending 0 |
| GICR_NSACR | 32 | 0 | rw | Interrupt Non-secure Access Control |
| GICR_PENDBASER | 64 | 0 | rw | Redistributor LPI Pending Table Base Address |
| GICR_PIDR0 | 32 | 93 | r- | Redistributor Peripheral ID 0 |
| GICR_PIDR1 | 32 | b4 | r- | Redistributor Peripheral ID 1 |
| GICR_PIDR2 | 32 | 3b | r- | Redistributor Peripheral ID 2 |
| GICR_PIDR3 | 32 | 0 | r- | Redistributor Peripheral ID 3 |
| GICR_PIDR4 | 32 | 44 | r- | Redistributor Peripheral ID 4 |
| GICR_PIDR5 | 32 | 0 | r- | Redistributor Peripheral ID 5 |
| GICR_PIDR6 | 32 | 0 | r- | Redistributor Peripheral ID 6 |
| GICR_PIDR7 | 32 | 0 | r- | Redistributor Peripheral ID 7 |
| GICR_PROPBASER | 64 | 0 | rw | Redistributor Properties Base Address |

| | | | | |
|---|---|---|---|---|
| GICR_SETLPIR | 64 | 0 | -w | Set LPI Pending |
| GICR_STATUSR | 32 | 0 | rw | Redistributor Status |
| GICR_SYNCR | 32 | 0 | r- | Redistributor Synchronize |
| GICR_TYPER | 64 | 9 | r- | Redistributor Type |
| GICR_WAKER | 32 | 6 | rw | Redistributor Wake |

Table 13.8: Registers at level 3, type:CPU group:MPCore_physical_redistributor

### 13.3.9 MPCore_processor_interface

Registers at level:3, type:CPU group:MPCore_processor_interface

| Name | Bits | Initial-Hex | RW | Description |
|---|---|---|---|---|
| GICC_ABPR | 32 | 3 | rw | Aliased Binary Point |
| GICC_AEOIR | 32 | 0 | -w | Aliased End of Interrupt |
| GICC_AHPPIR | 32 | 3ff | r- | Aliased Highest Priority Pending Interrupt |
| GICC_AIAR | 32 | 3ff | r- | Aliased Interrupt Acknowledge |
| GICC_APR0 | 32 | 0 | rw | Active Priorities 0 |
| GICC_BPR | 32 | 2 | rw | Binary Point |
| GICC_CTLR | 32 | 0 | rw | CPU Interface Control |
| GICC_DIR | 32 | 0 | -w | Deactivate Interrupt |
| GICC_EOIR | 32 | 0 | -w | End of Interrupt |
| GICC_HPPIR | 32 | 3ff | r- | Highest Priority Pending Interrupt |
| GICC_IAR | 32 | 3ff | r- | Interrupt Acknowledge |
| GICC_IIDR | 32 | 4043b | r- | CPU Interface ID |
| GICC_NSAPR0 | 32 | 0 | rw | Non-secure Active Priorities 0 |
| GICC_PMR | 32 | 0 | rw | Interrupt Priority Mask |
| GICC_RPR | 32 | ff | r- | Running Priority |
| GICC_STATUSR | 32 | 0 | rw | CPU Interface Status |

Table 13.9: Registers at level 3, type:CPU group:MPCore_processor_interface

### 13.3.10 MPCore_virtual_interface_control

Registers at level:3, type:CPU group:MPCore_virtual_interface_control

| Name | Bits | Initial-Hex | RW | Description |
|---|---|---|---|---|
| GICH_APR0 | 32 | 0 | rw | Active Priorities 0 |
| GICH_EISR0 | 32 | 0 | r- | End of Interrupt Status 0 |
| GICH_ELRSR0 | 32 | f | r- | Empty List Register Status 0 |
| GICH_HCR | 32 | 0 | rw | Hypervisor Control |
| GICH_LR0 | 32 | 0 | rw | List 0 |
| GICH_LR1 | 32 | 0 | rw | List 1 |
| GICH_LR2 | 32 | 0 | rw | List 2 |
| GICH_LR3 | 32 | 0 | rw | List 3 |
| GICH_MISR | 32 | 0 | r- | Maintenance Interrupt Status |
| GICH_VMCR | 32 | 4c0000 | rw | Virtual Machine Control |
| GICH_VTR | 32 | 90000003 | r- | VGIC Type |

Table 13.10: Registers at level 3, type:CPU group:MPCore_virtual_interface_control

### 13.3.11 MPCore_virtual_processor_interface

Registers at level:3, type:CPU group:MPCore_virtual_processor_interface

| Name | Bits | Initial-Hex | RW | Description |
|------|------|-------------|-----|-------------|
| GICV_ABPR | 32 | 3 | rw | VM Aliased Binary Point |
| GICV_AEOIR | 32 | 0 | -w | VM Aliased End of Interrupt |
| GICV_AHPPIR | 32 | 3ff | r- | VM Aliased Highest Priority Pending Interrupt |
| GICV_AIAR | 32 | 3ff | r- | VM Aliased Interrupt Acknowledge |
| GICV_APR0 | 32 | 0 | rw | VM Active Priorities 0 |
| GICV_BPR | 32 | 2 | rw | VM Binary Point |
| GICV_CTLR | 32 | 0 | rw | Virtual Machine Control |
| GICV_DIR | 32 | 0 | -w | VM Deactivate Interrupt |
| GICV_EOIR | 32 | 0 | -w | VM End of Interrupt |
| GICV_HPPIR | 32 | 3ff | r- | VM Highest Priority Pending Interrupt |
| GICV_IAR | 32 | 3ff | r- | VM Interrupt Acknowledge |
| GICV_IIDR | 32 | 4043b | r- | VM CPU Interface ID |
| GICV_PMR | 32 | 0 | rw | VM Priority Mask |
| GICV_RPR | 32 | ff | r- | VM Running Priority |
| GICV_STATUSR | 32 | 0 | rw | VM CPU Interface Status |
| ICV_AP0R0_EL1 | 32 | 0 | rw | VM Group 0 Active Priorities 0 |
| ICV_AP1R0_EL1 | 32 | 0 | rw | VM Group 1 Active Priorities 0 |
| ICV_BPR0_EL1 | 32 | 2 | rw | VM Group 0 Binary Point |
| ICV_BPR1_EL1 | 32 | 3 | rw | VM Group 1 Binary Point |
| ICV_CTLR_EL1 | 32 | 400 | rw | Virtual Machine Control |
| ICV_DIR_EL1 | 32 | 0 | -w | VM Deactivate Interrupt |
| ICV_EOIR0_EL1 | 32 | 0 | -w | VM Group 0 End of Interrupt |
| ICV_EOIR1_EL1 | 32 | 0 | -w | VM Group 1 End of Interrupt |
| ICV_HPPIR0_EL1 | 32 | 3ff | r- | VM Group 0 Highest Priority Pending Interrupt |
| ICV_HPPIR1_EL1 | 32 | 3ff | r- | VM Group 1 Highest Priority Pending Interrupt |
| ICV_IAR0_EL1 | 32 | 3ff | r- | VM Group 0 Interrupt Acknowledge |
| ICV_IAR1_EL1 | 32 | 3ff | r- | VM Group 1 Interrupt Acknowledge |
| ICV_IGRPEN0_EL1 | 32 | 0 | rw | VM Group 0 Interrupt Enable |
| ICV_IGRPEN1_EL1 | 32 | 0 | rw | VM Group 1 Interrupt Enable |
| ICV_PMR_EL1 | 32 | 0 | rw | VM Priority Mask |
| ICV_RPR_EL1 | 32 | ff | r- | VM Running Priority |

Table 13.11: Registers at level 3, type:CPU group:MPCore_virtual_processor_interface

### 13.3.12   MPCore_ITS

Registers at level:3, type:CPU group:MPCore_ITS

| Name | Bits | Initial-Hex | RW | Description |
|------|------|-------------|-----|-------------|
| GITS_BASER0 | 64 | 0 | rw | ITS Translation Table Descriptor 0 |
| GITS_BASER1 | 64 | 0 | rw | ITS Translation Table Descriptor 1 |
| GITS_BASER2 | 64 | 0 | rw | ITS Translation Table Descriptor 2 |
| GITS_BASER3 | 64 | 0 | rw | ITS Translation Table Descriptor 3 |
| GITS_BASER4 | 64 | 0 | rw | ITS Translation Table Descriptor 4 |
| GITS_BASER5 | 64 | 0 | rw | ITS Translation Table Descriptor 5 |
| GITS_BASER6 | 64 | 0 | rw | ITS Translation Table Descriptor 6 |
| GITS_BASER7 | 64 | 0 | rw | ITS Translation Table Descriptor 7 |
| GITS_CBASER | 64 | 0 | rw | ITS Command Queue Descriptor |
| GITS_CREADR | 64 | 0 | rw | ITS Read |
| GITS_CTLR | 32 | 80000000 | rw | ITS Control |
| GITS_CWRITER | 64 | 0 | rw | ITS Write |
| GITS_IIDR | 32 | 43b | r- | ITS Identification |
| GITS_TYPER | 64 | 9ef79 | r- | ITS Type |

Table 13.12: Registers at level 3, type:CPU group:MPCore_ITS

Copyright (c) 2024 Imperas Software Limited     www.ovpworld.org

OVP License. Release 20240902.0     Page 56 of 56