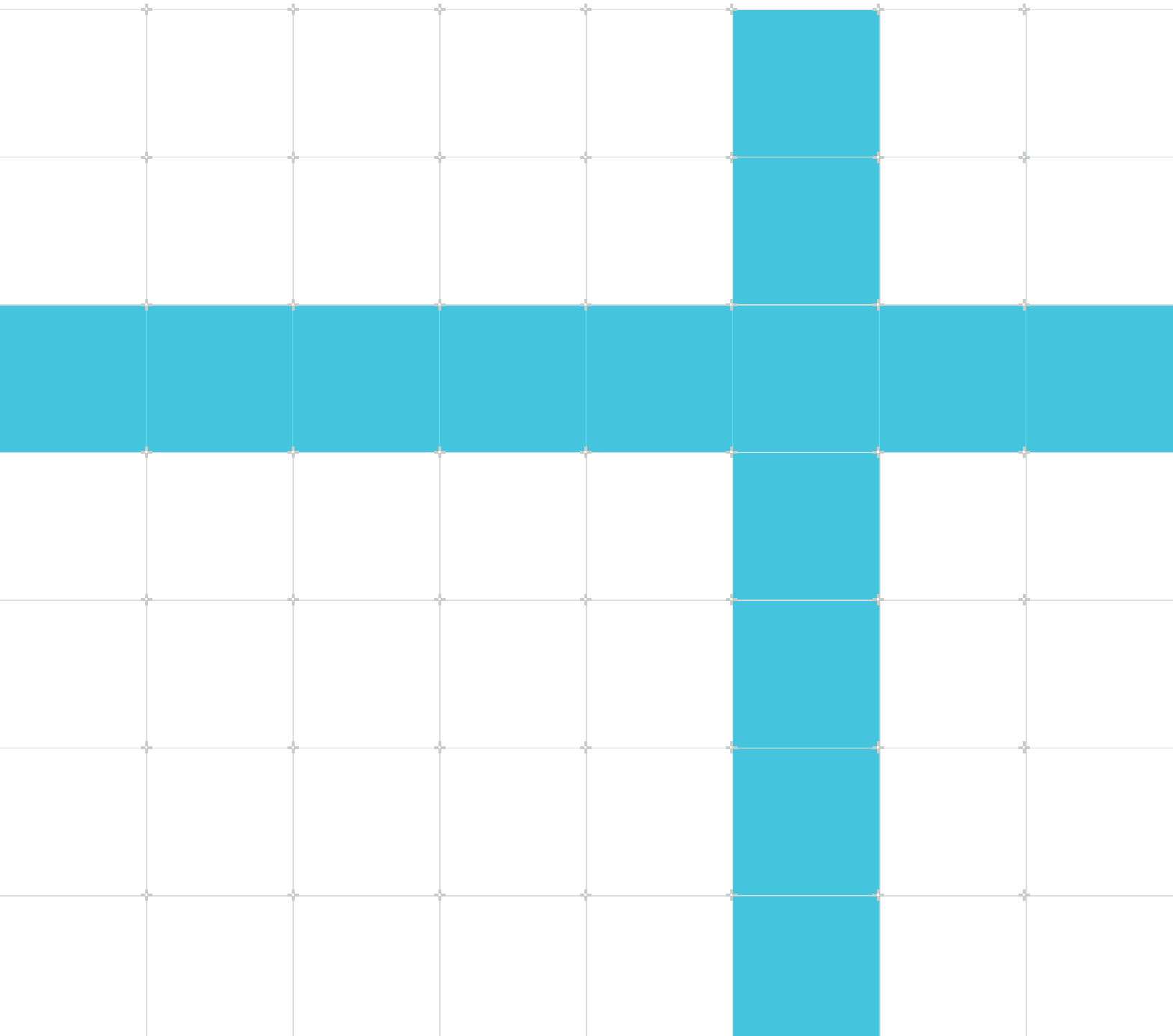
Arm® CoreSight™ System-on-Chip SoC-600

Revision r7p1

Technical Reference Manual

Non-Confidential

Issue 17
100806_0701_17_en



Arm® CoreSight™ System-on-Chip SoC-600 Technical Reference Manual

This document (100806_0701_17_en) was issued on 2025-06-30. There might be a later issue at https://developer.arm.com/documentation/100806

The product revision is r7p1.

See also: Proprietary notice | Product and document information | Useful resources

Start reading

If you prefer, you can skip to the start of the content.

Intended audience

This document is written for hardware and software engineers who want to incorporate CoreSight™ SoC-600 into their design and produce real-time instruction and data trace information from a SoC.

Inclusive language commitment

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

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

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

Feedback

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

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

Contents

1. About CoreSight SoC-600

CoreSight SoC-600 is a member of the Arm embedded debug and trace component family. Hence, SoC-600 provides components that can be used for debug and trace of Arm SoCs. These SoCs can be simple single-processor designs to complex multiprocessor and multi-cluster designs that include many heterogeneous processors.

The components:

- Support for the Arm Debug Interface (ADI) v6 and CoreSight™ v3 Architectures that enable you to build debug and trace functionality into your systems. It supports debug and trace over existing functional interfaces.
- Components that support the development of low-power system implementations through architected fine-grained power control.
- Q-Channel interfaces for clock and power quiescence.
- Can be integrated with the Arm CoreLink™ LPD-500 as part of a full-chip power and clock control methodology.
- The Arm CoreSight SDC-600 can be integrated with CoreSight SoC-600, with an applicable license, as part of a certificate-based authenticated debug solution.

The CoreSight SoC-600 package includes:

- A library of configurable CoreSight components that are written in Verilog, and that are compliant with the Verilog-2001 Standard IEEE Std 1364-2001.
- Example timing constraint files for each component in SDC format.


Note

Regarding Arm® Socrates™ IP Tooling:

- CoreSight SoC-600 was previously configurable in Socrates System Builder.
- Socrates System Builder is now in maintenance. There will be no functionality updates to Socrates System Builder, although there might be maintenance updates.
- The IP Tooling platform, Socrates, enables you to configure and build individual CoreSight components. However, there is no CoreSight creation flow, and no system-stitching capability.
- You can enable Socrates functionality using a legacy Socrates System Builder license.

1.1 Features

SoC-600 provides debug, trace, embedded cross triggering, and power features and capabilities.

The key features of SoC-600 are:

Debug

- *Arm® Debug Interface Architecture Specification ADIv6.0*-compliant debug port. This debug port supports JTAG and Serial Wire protocols for connection to an off-chip debugger. This connection is achieved using a low-pin-count connection that is suitable for bare-metal debug and silicon bring-up.
- *Arm® CoreSight™ Architecture Specification v3.0* compliance enables debug over functional interfaces, suitable for application development and in-field debug without a dedicated debug interface.
- Infrastructure components supporting system identification and integration with other CoreSight IP.

Trace

- Versatile Trace Memory Controller (TMC) supporting local on-chip storage, and buffering of trace data.
- TMC router configuration supports efficient hand-off of trace data to other system AXI managers. This feature enables trace over functional interfaces, suitable for application development and in-field debug without a dedicated debug and trace interface.
- TMC streaming configuration supports integration to third-party High Speed Serial Trace Ports (HSSTP) for high bandwidth, low pin count trace solutions.
- Infrastructure components supporting filtering and routing of trace data on chip.

Embedded Cross Triggering

- Cross Trigger Interface (CTI) supports up to 32 trigger inputs and outputs with a single component instance.
- Cross Trigger Matrix (CTM) supports up to 33 CTI or CTM connections without cascading.
- Cross Trigger components support 4 or 16 channels.

Power

- *Arm® CoreSight™ Architecture Specification v3.0*-compliant Granular Power Requester (GPR) enables fine-grained debug and system power control at all levels of debug hierarchy.
- Components are designed for low-power implementation, supporting clock and power quiescence and wakeup signaling where necessary.
- Components support Q-Channel Low-Power Interfaces (LPI) for integration with power controllers to support system-level clock and power gating where necessary.
- Infrastructure components support implementation across multiple clock and power domains.

Miscellaneous

- Some components, such as the bridges and Serial Wire Debug Port (SW-DP), use two Verilog modules to span clock and power domains. This design can ease implementation in complex SoC designs that have multiple clock and power domains.
- Infrastructure components support integration with legacy IP including Arm CoreSight Architecture Specification v2.0-compliant, and JTAG components.

1.2 Supported standards

CoreSight SoC-600 is compliant with a mixture of Arm specifications and external standards.

The specifications and standards are as follows:

- *Arm® CoreSight™ Architecture Specification v3.0*
- *Arm® Debug Interface Architecture Specification ADIv6.0*
- *AMBA® APB Protocol Specification*
- *AMBA® ATB Protocol Specification*
- *AMBA® AHB Protocol Specification*
- *AMBA® AXI-Stream Protocol Specification*
- *AMBA® AXI Protocol Specification*
- *AMBA® Low Power Interface Specification*
- *Arm® Embedded Trace Macrocell Architecture Specification ETMv4.0 to ETMv4.3*
- *CoreSight™ Program Flow Trace Architecture Specification PFTv1.0 and PFTv1.1*
- *Verilog-2001 Standard*
- *Accellera IP-XACT version 1685-2009*
- *IEEE Standard Test Access Port and Boundary Scan Architecture (JTAG)*

1.3 Documentation

The SoC-600 documentation includes a Technical Reference Manual (TRM) and a Configuration and Integration Manual (CIM). These documents relate to the SoC-600 design flow.

Technical Reference Manual

The TRM describes the functionality and the effects of functional options on the behavior of the SoC-600 components. It is required at all stages of the design flow. The choices that you make in the design flow can mean that some behavior that is described in the TRM is not relevant. If you are programming a device that is based on SoC-600 components, then contact the integrator to determine the configuration of your device.

Configuration and Integration Manual

The CIM describes:

- How to configure the SoC-600 components.
- How to integrate the SoC-600 components into your SoC design and how to configure system-specific Identification Registers.
- How to implement the SoC-600 components to produce a hard macrocell of the design.

This description includes custom cell replacement, a description of the power domains, and a description of the design synthesis.

The CIM is a confidential book that is only available to licensees.

1.4 Design process

The SoC-600 components are delivered as synthesizable Verilog RTL.

Before the SoC-600 components can be used in a product, they must go through the following processes:

System design

Determining the necessary structure and interconnections of the SoC-600 components that form the CoreSight™ debug and trace subsystem.

System configuration

Defining the memory map of the system and the functional configuration of the SoC-600 components.

Integration

Connecting the SoC-600 components together, and to the SoC memory system and peripherals.

Verification

Verifying that the CoreSight debug and trace subsystem has been correctly integrated to the processor or processors in your SoC.

Implementation

Using the Verilog RTL in an implementation flow to produce a hard macrocell. The operation of the final device depends on:

Component configuration

The implementer chooses the options that affect how the RTL source files are pre-processed. These options usually include, or exclude, logic that affects one or more of the area, maximum frequency, and features of the resulting macrocell.

Software configuration

The programmer configures the CoreSight debug and trace subsystem by programming specific values into registers that affect the behavior of the SoC-600 components.

Note

SoC-600 is highly configurable to support many system topologies. We recommend that you follow the guidance in the *Arm® CoreSight™ Base System Architecture*. This ensures wide support for your product across the Arm debug ecosystem.

1.5 Component list

CoreSight SoC-600 components are provided as RTL blocks. Every block name is prefixed with `css600_`.

The following table shows the components and their versions. The Revision column only applies to components with a programmers model. The Revision value is for the PIDR2.REVISION field.

Table 1-1: SoC-600 component list

Name	Description	Version	Revision	IP-XACT version
`css600_ahbap`	AHB Access Port	r3p0	5	r3p0_1
`css600_apbap`	APB Access Port	r2p0	4	r2p0_1
`css600_apbasyncbridge`	APB Asynchronous Bridge	r1p1	-	r1p1_0
`css600_apbic`	APB Interconnect	r1p0	-	r1p0_1
`css600_apbpaddrdbg31adapter`	APB PADDRDBG[31] Adapter	r2p0	-	r2p0_1
`css600_apbrom`	APB ROM Table	r1p1	0	r1p1_0
`css600_apbrom_gpr`	APB ROM Table with Granular Power Requester	r1p1	0	r1p1_0
`css600_apbsyncbridge`	APB Synchronous Bridge	r2p0	-	r2p0_0
`css600_apv1adapter`	Access Port v1 Adapter	r1p0	1	r1p0_1
`css600_atbasyncbridge`	ATB Asynchronous Bridge	r2p1	-	r2p1_0
`css600_atbbuffer`	ATB Trace Buffer	r1p1	-	r1p1_0
`css600_atbdownsizer`	ATB Downsizer	r1p1	-	r1p1_0
`css600_atbfunnel`	ATB Trace Funnel	r1p1	5	r1p1_0
`css600_atbfunnel_prog`	ATB Programmable Trace Funnel	r1p1	5	r1p1_0
`css600_atbreplicator`	ATB Trace Replicator	r1p1	5	r1p1_0
`css600_atbreplicator_prog`	ATB Programmable Trace Replicator	r1p1	5	r1p1_0
`css600_atbsyncbridge`	ATB Synchronous Bridge	r2p1	-	r2p1_0
`css600_atbupsizer`	ATB Trace Upsizer	r1p1	-	r1p1_0
`css600_authasyncbridge`	Authentication Asynchronous Bridge	r1p0	-	r1p0_1
`css600_authreplicator`	Authentication Replicator	r1p0	-	r1p0_1
`css600_authsyncbridge`	Authentication Synchronous Bridge	r1p0	-	r1p0_1
`css600_axiap`	AXI Access Port	r4p0	6	r4p0_2
`css600_axiap_mte`	AXI Access Port with MTE	r4p0	6	r4p0_2
`css600_catu`	CoreSight Address Translation Unit	r1p0	2	r1p0_2
`css600_channelpulseasyncbridge`	Channel Pulse Asynchronous Bridge	r1p0	-	r1p0_1
`css600_channelpulsesyncbridge`	Channel Pulse Synchronous Bridge	r1p0	-	r1p0_1
`css600_channelpulsetochanneladapter`	Channel Pulse to Channel Adapter	r1p0	-	r1p0_1
`css600_channeltochannelpulseadapter`	Channel to Channel Pulse Adapter	r1p0	-	r1p0_1
`css600_cortexa5integrationcs`	Cortex-A5 PIL	r1p0	2	r1p0_1
`css600_cortexa8integrationcs`	Cortex-A8 PIL	r1p0	1	r1p0_0

Name	Description	Version	Revision	IP-XACT version
css600_cortexa9integrationcs	Cortex-A9 PIL	r1p0	1	r1p0_0
css600_cortexm0integrationcs	Cortex-M0 PIL	r1p0	1	r1p0_1
css600_cortexm3integrationcs	Cortex-M3 PIL	r1p0	1	r1p0_1
css600_cortexm4integrationcs	Cortex-M4 PIL	r1p0	1	r1p0_1
css600_cortexr4integrationcs	Cortex-R4 PIL	r1p0	2	r1p0_0
css600_cortexr5integrationcs	Cortex-R5 PIL	r1p0	2	r1p0_0
css600_cti	Cross Trigger Interface	r1p0	4	r1p0_2
css600_ctitostmadapter	CTI to STM Adapter	r1p0	-	r1p0_1
css600_ctm	Cross Trigger Matrix	r1p0	-	r1p0_1
css600_dp	Debug Port	r1p0	5	r1p0_2
css600_dpabortasyncbridge	DP Abort Asynchronous Bridge	r1p0	-	r1p0_1
css600_dpabortreplicator	DP Abort Replicator	r1p0	-	r1p0_1
css600_dpabortsyncbridge	DP Abort Synchronous Bridge	r1p0	-	r1p0_1
css600_eventlevelasyncbridge	Event Level Asynchronous Bridge	r1p0	-	r1p0_1
css600_eventlevelsyncbridge	Event Level Synchronous Bridge	r1p0	-	r1p0_1
css600_eventpulseasyncbridge	Event Pulse Asynchronous Bridge	r1p0	-	r1p0_1
css600_eventpulsesyncbridge	Event Pulse Synchronous Bridge	r1p0	-	r1p0_1
css600_eventpulsetoeventadapter	Event Pulse to Event Adapter	r1p0	-	r1p0_1
css600_eventtoeventpulseadapter	Event to Event Pulse Adapter	r1p0	-	r1p0_1
css600_jtagap	JTAG Access Port	r1p0	4	r1p0_1
css600_jtagtoswjadapter	JTAG to SWJ Adapter	r1p0	-	r1p0_1
css600_swjic	SWJ Interconnect	r2p0	-	r2p0_1
css600_swjtojtagadapter	SWJ to JTAG Adapter	r1p0	-	r1p0_1
css600_tmc_etb	Trace Memory Controller - Embedded Trace Buffer	r3p0	9	r3p0_1
css600_tmc_etf	Trace Memory Controller - Embedded Trace FIFO	r3p0	9	r3p0_1
css600_tmc_etr	Trace Memory Controller - Embedded Trace Router	r3p0	9	r3p0_1
css600_tmc_ets	Trace Memory Controller - Embedded Trace Streamer	r3p0	9	r3p0_1
css600_tpiu	Trace Port Interface Unit	r2p1	4	r2p1_0
css600_tsgen	Timestamp Generator	r1p0	0	r1p0_1
css600_tsintp	Timestamp Interpolator	r1p0	-	r1p0_1
css600_tsreplicator	Timestamp Replicator	r1p0	-	r1p0_1

1.6 Product revisions

The SoC-600 functionality may change between product revisions.

r0p0

First release of CoreSight SoC-600.

r1p0

Added the following components:

- Narrow Timestamp (NTS) components. Timestamp interpolator.
- Processor Integration Layer (PIL) component for the Arm® Cortex®-M4 processor.

r2p0

Added the following components:

- CoreSight™ Address Translation Unit (CATU).
- PIL components for the Cortex-M0, Cortex-M3, Cortex-R4, Cortex-R5, Cortex-A5, Cortex-A8, and Cortex-A9 processors.

r3p0

Added the Trace Port Interface Unit (TPIU) component.

r3p1

Component errata fixes.

r3p2

Component errata fixes.

r4p0

The functional changes are:

- AHB-AP updated to allow a debugger to have greater control over the AHB transaction attributes, which is required for the latest Cortex-M processors. This change updates the programmers model.
- AXI-AP updated to support the Memory Tagging Extension (MTE). This update enables a debugger to read and write memory tags with the memory data. MTE support is a configuration option and extends the AXI-AP programmers model.

r4p1

Component errata fixes.

r4p2

Component errata fixes.

r5p0

The functional changes are:

- APB-AP and AXI-AP now support the Realm Management Extension (RME). This update enables the debugger to generate root and realm transactions when permitted by the authentication signals.
- AHB-AP, APB-AP, and AXI-AP support separate authentication control signals for the two internal logical-APs. APB infrastructure components support physical address space signaling.
- APB ROM table configuration supports 256 64-bit entries or 512 32-bit entries (default).
- TMC Embedded Trace Buffer (ETB) and Embedded Trace FIFO (ETF) have an MBIST interface. TMC Embedded Trace Router (ETR) has improved AXI performance.
- Embedded Cross Trigger (ECT) components support 4 (default) or 16 channels.

- Narrow Timestamp (NTS) components are removed.

r6p0

The functional changes are:

- The AXI-AP and AXI-AP with MTE (`css600_axiap` and `css600_axiap_mte`) are updated to support the Memory Encryption Contexts Identifier (MECID) feature of the AMBA AXI protocol, issue K. See Memory Encryption Contexts ID.
- TMC ETR translation stashing. The Trace Memory Controller Embedded Trace Router (`css600_tmc_etr`) is updated to improve performance when used with a compatible System Memory Management Unit (SMMU). See AXI manager interface and the AXICTL1 register.

r7p0

The functional changes are:

- TMC ETR translation stashing. The Trace Memory Controller Embedded Trace Router (`css600_tmc_etr`) is updated to support the unstash command for use with compatible System Memory Management Units (SMMUs).

r7p1

Component errata fixes.

2. DAP components functional description

The SoC-600 Debug Access Port (DAP) components include ports, bridges, and interfaces.

2.1 Debug port

The SoC-600 `css600_dp` module implements the JTAG and Serial Wire Debug Port protocols. You can omit either of these protocols to save area in systems that do not require both protocols.

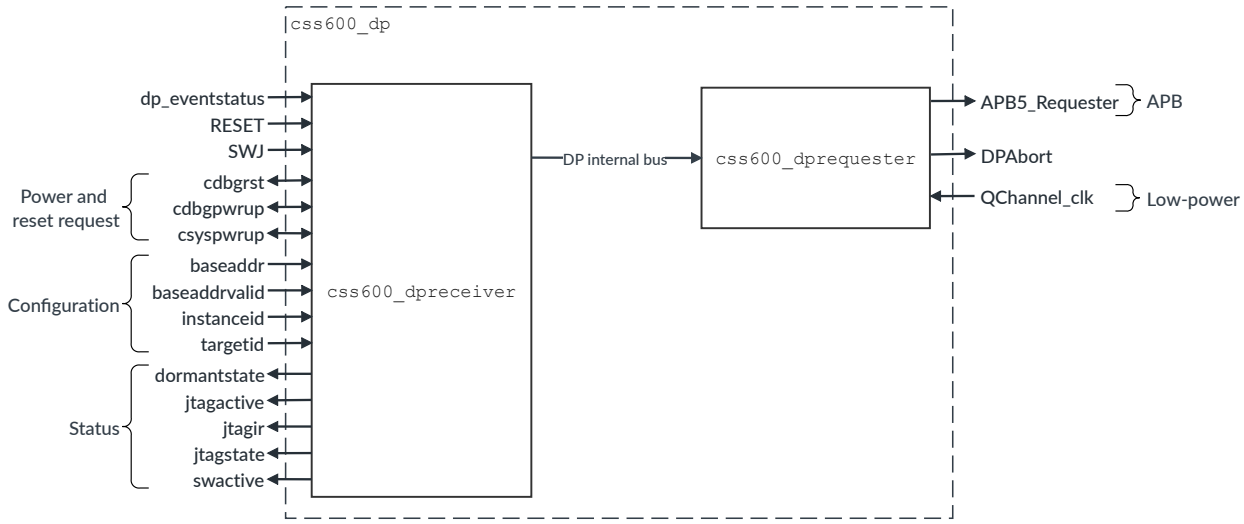
The debug port communicates with the debug components through the APB infrastructure that is connected to the debug port APB requester interface.

The debug port implements the following features:

- ADIv6 architecture
- Single clock domain in each part
- Asynchronous bridge between the completer and requester parts
- 4-bit or 8-bit Instruction register for JTAG implementation
- Separate completer and requester components, implementing JTAG, Serial Wire, or both in the completer, and APB in the requester

The following figure shows the external connections on the Debug Port (DP).

Figure 2-1: css600_dp logical connections



2.2 Memory access ports

Memory Access Ports connect one memory system to another using one of the AMBA® bus protocols. The AMBA bus protocols used by SoC-600 are AHB, APB, and AXI.

The *Arm® Debug Interface Architecture Specification ADIv6.0* defines a Memory Access Port (MEM-AP) so that it provides two logical views of the access port to the debugger. These two views are referred to as twin APs or logical APs.

In SoC-600, these two logical APs are contiguous in the memory map and each one of them occupies 4kB address space. An external debugger can only discover one of the twin APs through the ROM table. The other AP is dedicated for self-hosted debug.

The MEM-AP itself is not capable of differentiating which of the twin APs is visible in the ROM table. The MEM-AP decodes the access requests on the APB completer interface and maps them to AP-L0 or AP-L1, based on the value of the paddr_s[12] signal.

The following table shows the implementation-defined features of MEM-APs that SoC-600 supports.

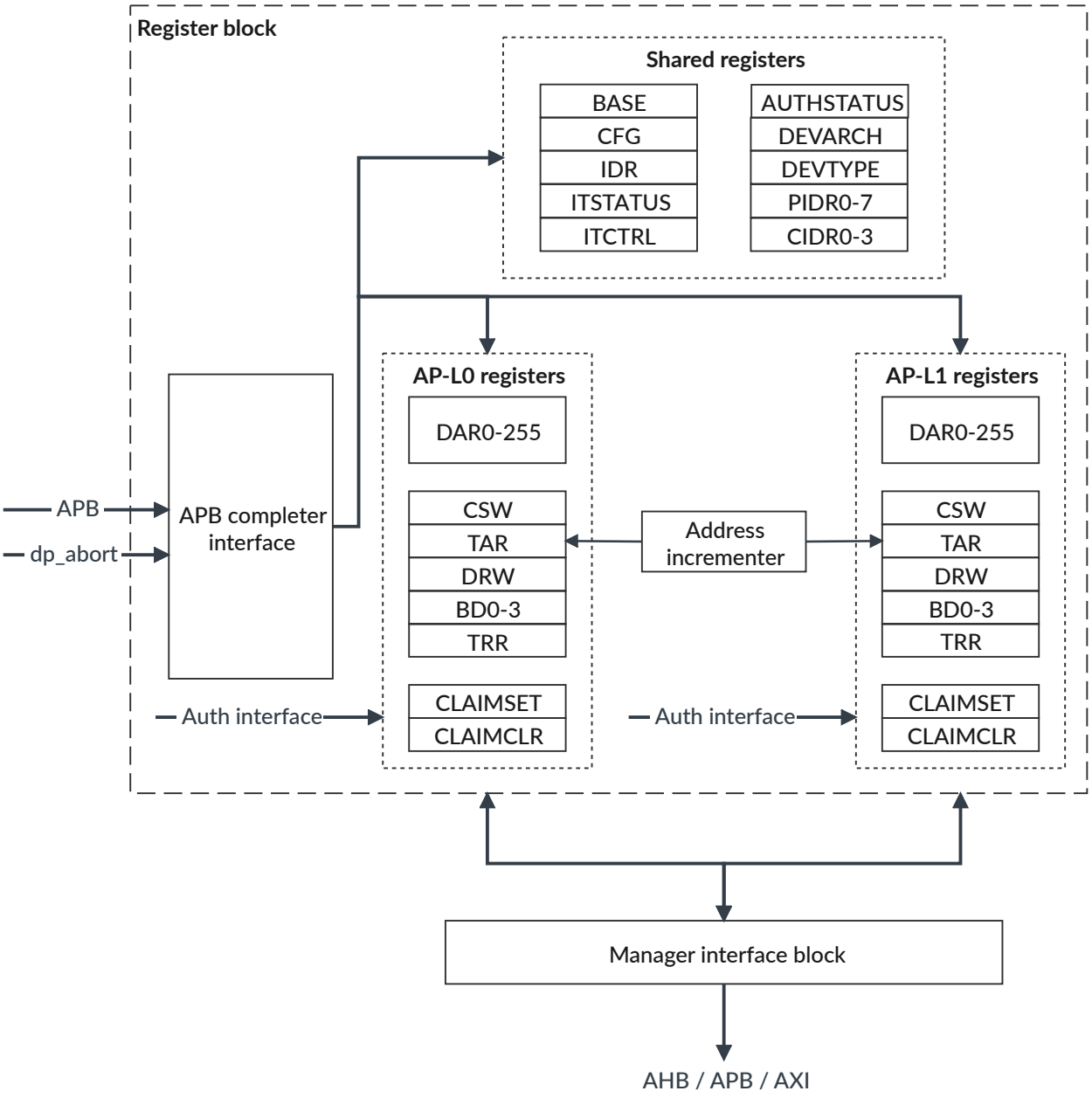
Table 2-1: MEM-AP implementation-defined features

Feature	AHB-AP	APB-AP	AXI-AP	AXI-AP with MTE
Packed transfers	No	No	No	No
Non-word sizes (smaller than 32-bit)	Supported	No	Supported	Supported
Large Data Extension (64-bit)	No	No	Configurable option	Yes
Large Physical Address Extension (64-bit)	No	No	Configurable option	Yes
Memory Tagging Extension, MTE	No	No	No	Yes
Separate authentication control signals	Configurable option	Configurable option	Configurable option	Configurable option
Realm Management Extension (RME)	No	Yes	Yes	Yes

The completer and requester interfaces are shared by both, AP-L0 and AP-L1. They also share all read-only registers, as the following figure shows. Therefore, the read value returned by these registers is the same, irrespective of whether they are accessed through AP-L0 or AP-L1. However, the writeable registers are duplicated on both logical APs. Accessing them in one view does not affect the state in the other view.

The following diagram shows the MEM-AP block diagram.

Figure 2-2: MEM-AP block diagram



2.2.1 APB Access Port

The SoC-600 `css600_apbap` component is a Memory Access Port (MEM-AP). The `css600_apbap` is an AMBA® APB5 completer component that provides access to another APB5 memory system.

Use the `css600_apbap` component to provide access to an APB5 memory space, for example:

- A subsystem of CoreSight™ components that includes Arm® Cortex®-A or Cortex-R processors
- A subsystem of CoreSight components

- Any other APB5 memory system

The APB Access Port allows visibility into another memory system from the debug APB infrastructure. Access Ports and related infrastructure can be cascaded in a CoreSight system to any depth. This process allows any memory system to contain a window into another memory system with a maximum memory footprint of 8KB in the source memory system.

The APB-AP provides an APB5 completer interface for programming and an APB5 requester interface for accessing the target memory system. The programmers model contains the details of the registers for accessing the features of the APB5 requester interface.

The APB-AP provides the following features:

- Error response
- Stalling accesses
- Little-endian only
- Single clock domain
- 32 bits data access only
- Auto-incrementing Transfer Address Register (TAR)
- An APB5 completer interface
- An APB5 requester interface
- An Access Port Enable interface
- CoreSight Component base pointer register
- A Q-Channel LPI for high-level clock management
- Realm Management Extension (RME) support

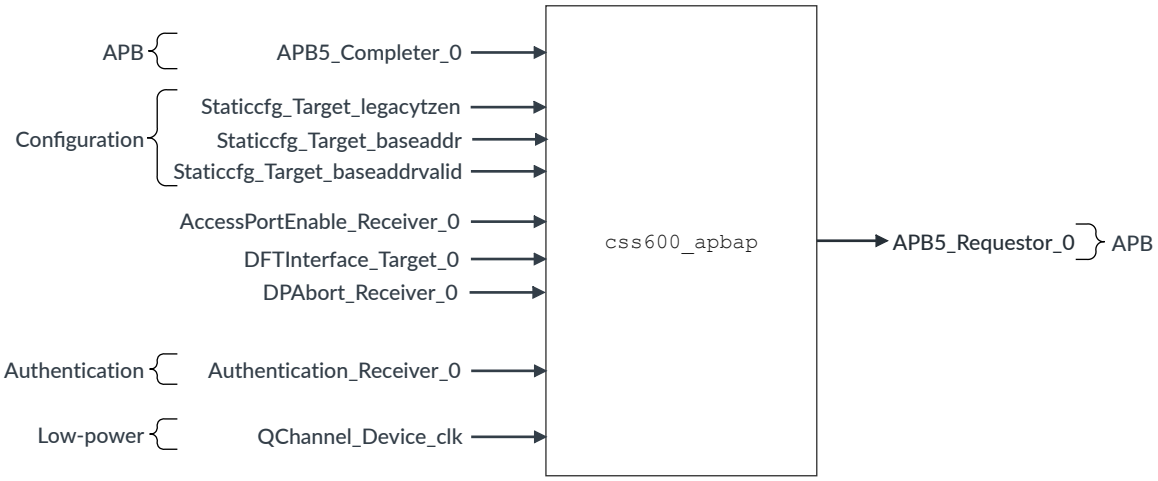
The APB-AP does not support subword transfers.


Note

If the DP issues an abort over the Debug APB interface, the APB-AP completes the transaction on its Debug APB completer interface immediately. The DP transfer abort does not cancel the ongoing APB transfer on the APB requester interface.

The following figure shows the external connections on the APB Access Port.

Figure 2-3: css600_apbap logical connections



2.2.2 AHB Access Port

The SoC-600 `css600_ahbap` component is a Memory Access Port (MEM-AP). The `css600_ahbap` is an AMBA® APB5 completer component that provides access to an AMBA AHB5 memory system.

Use the `css600_ahbap` to provide access to an AHB5 memory space, for example:

- An Arm® Cortex®-M processor and subsystem
- Any other AHB5 memory system

The AHB Access Port allows visibility into another memory system from the debug APB infrastructure. Access Ports and related infrastructure can be cascaded in a CoreSight™ system to any depth. This process allows any memory system to contain a window into another memory system with a maximum memory footprint of 8KB in the source memory system.

The AHB-AP provides an APB5 completer interface for programming and an AHB5 manager interface for accessing the target memory system. The programmers model contains the details of the registers for accessing the features of the AHB manager interface.

The AHB-AP provides the following features:

- Error response
- Stalling accesses
- Little-endian only
- Single clock domain
- Auto-incrementing Transfer Address Register (TAR)
- An APB5 completer interface
- An AHB5 manager interface
- An Access Port Enable interface

- 8 bits, 16 bits, or 32 bits data access
- CoreSight Component base pointer register
- Support for AHB5 TrustZone® signaling
- A Q-Channel LPI for high-level clock management

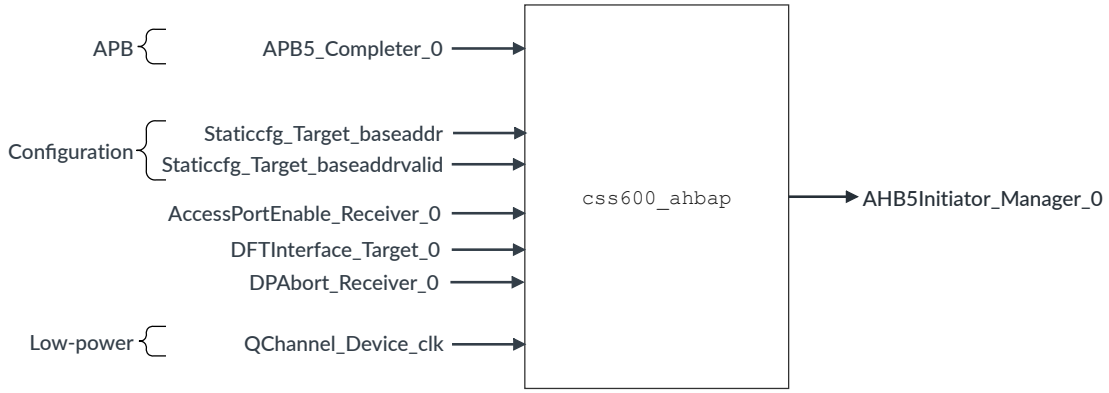
The AHB-AP does not support:

- Exclusive accesses
- Unaligned transfers
- BURST or SEQ transactions

If the DP issues an abort to AHB-AP, the AHB-AP completes the transaction on its APB completer interface immediately. The DAP transfer abort does not cancel the ongoing AHB transfer.

The following figure shows the external connections on the AHB Access Port.

Figure 2-4: css600_ahbap logical connections



2.2.3 AXI Access Port

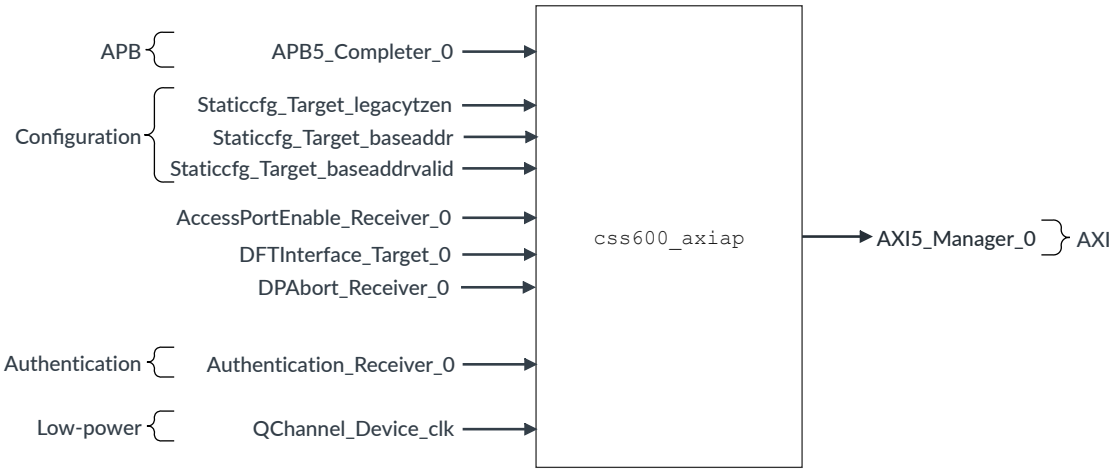
The AXI Access Port implements the MEM-AP architecture to connect directly to an AXI memory system. You can connect it to other memory systems using a suitable bridging component.

There are two AXI Access Ports:

- `css600_axiap_mte` with Memory Tagging Extension (MTE)
- `css600_axiap` without MTE

The following figure shows the external connections on the AXI Access Port.

Figure 2-5: css600_axiap logical connections



2.2.3.1 AXI-AP features

The SoC-600 AXI Access Port (AXI-AP) implements the following features:

- Error response
- Stalling accesses
- Little-endian only
- Single clock domain
- AXI5 interface support
- Memory Tagging Extension (MTE)
- Memory Encryption Contexts ID (MECID)
- An Access Port Enable interface
- Auto-incrementing Transfer Address Register (TAR)
- 8, 16, 32, or 64 bits data access
- 32 bits or 64 bits Large Physical Address (LPA) extension support
- AXI transfers:
  - Burst size of 1 only
  - Write and read transfers
  - No out-of-order transactions
  - No multiple outstanding accesses
  - Only aligned transfers are supported

2.2.3.2 DAP transfer abort

If the DP issues an abort to the AXI-AP, the AXI-AP completes the transaction on its APB completer interface immediately. The DAP transfer abort does not cancel the ongoing AXI transfer.

2.2.3.3 Additional AXI error responses

The AXI-AP produces error responses for AXI-initiated and AP-initiated transfers.

AXI initiated error responses

If the AXI-AP receives an error response on the AXI manager interface, it propagates onto the APB bus as the transfer completes.

For a 64-bit data transfer, a sequence of two reads or writes must be generated on the APB completer interface for a single 64-bit access on the AXI interface. For reads, the first read request on the APB completer interface sends a read request on the AXI interface. For writes, the AXI-AP sends a write access on the AXI interface only after receives two write requests on the APB bus.

Therefore, an error response that the AXI-AP receives for a read request, is for the first read request on the APB bus. An error response that the AXI-AP receives for a write request, is for the second write request on the APB bus.

AP-initiated error response

- **AXI-AP reads after a 64-bit AXI read sequence is broken**

  Read requests from the APB bus of the AXI-AP must access both BDx registers, or a consecutive pair of DAR registers forming an aligned 64-bit address. Read requests must access the lower-numbered register first. For a DRW register access, two read requests are required to get the entire 64-bit word from the AXI interface.

  All other accesses, such as a read followed by a write access to the same or different registers, return an error response to the DP.

- **AXI-AP writes after a 64-bit write sequence is broken**

  Write requests from the APB interface of the AXI-AP must access both BDx registers of the pair, or consecutive DAR registers forming an aligned 64-bit address. They must access the lower-numbered register first. For a DRW register access, two write requests are required to build a 64-bit packet as write data on the AXI interface.

  All other accesses, such as a write followed by another read or write access to different registers, return an error response. For example, after accessing the DRW register, the next access on the Debug APB bus must be a write to the DRW register. Any other access returns an error response.

  Similarly, after accessing BD0, the next access must be a write to BD1. Any other access returns an error response.

2.2.3.4 AXI transfers

The AMBA® 5 AXI-compliant manager interface processes one transaction at a time.

The manager interface does not support:

- More than one transfer at a time
- Out-of-order transactions

Burst length

The AXI-AP supports a burst length of one transfer only. ARLEN[3:0] and AWLEN[3:0] signals are always `0b0000`.

Burst size

Supported burst sizes are:

- 8-bit
- 16-bit
- 32-bit
- 64-bit

Burst type

ARBURST[1:0] and AWBURST[1:0] signals are always `0b01`. Burst type has no meaning in this context because only bursts of one transfer are supported.

Atomic accesses

AXI-AP supports normal accesses only. ARLOCK and AWLOCK signals are always `0b0`.

Unaligned accesses

The AXI-AP does not support unaligned data accesses. If you attempt to read from or write to an address that is not aligned to the data width specified by CSW. Size, the lower address bits are driven with zeros.

Memory Tagging Extension (MTE) support

When using the `css600_axiap_mte` component, the AXI-AP supports reading (artagop_m=Transfer) and writing (awtagop_m=Update) memory tags with the memory data.

2.2.3.5 Valid combinations of AxCACHE and AxDOMAIN

Use valid combinations of AxCACHE and AxDOMAIN according to the access type required for the domain type.

The following table shows valid combinations of AxCACHE and AxDOMAIN.

Table 2-2: Valid combinations of AxCACHE and AxDOMAIN values

AxCACHE[3:0] CSW.Cache	Access type	AxDOMAIN CSW.Domain	Domain type	Valid CSW.MTE=0	Valid CSW.MTE=1
`0b000x`	Device	`0b00`	Non-shareable	No	No

AxCACHE[3:0] CSW.Cache	Access type	AxDOMAIN CSW.Domain	Domain type	Valid CSW.MTE=0	Valid CSW.MTE=1
0b000x	Device	0b01	Shareable	No	No
0b000x	Device	0b10	Shareable	No	No
0b000x	Device	0b11	System	Yes	No
0b001x	Normal Non-cacheable	0bxx	Any	Yes	No
0b010x	Reserved	0bxx	Any	No	No
0b100x	Reserved	0bxx	Any	No	No
0b110x	Reserved	0bxx	Any	No	No
0b011x	Normal Cacheable	0b00	Non-shareable	Yes	CSW.Cache[0]=1 : Yes CSW.Cache[0]=0 : No
0b101x	Normal Cacheable	0b01	Shareable	Yes	CSW.Cache[0]=1 : Yes CSW.Cache[0]=0 : No
0b111x	Normal Cacheable	0b10	Shareable	Yes	CSW.Cache[0]=1 : Yes CSW.Cache[0]=0 : No
0b111x	Normal Cacheable	0b11	System	No	No

2.2.3.6 AXI access generation

For the AXI-AP, you must set a valid combination of CSW.Size, CSW.MTE, CSW.Cache, and CSW.Domain. A valid combination must be set performing a read or write transaction on the AXI manager interface using DRW, BD, or DAR registers.

When setting CSW.Cache and CSW.Domain, see Valid combinations of AxCACHE and AxDOMAIN.

The following table shows the other conditions required to generate an AXI access for the `css600_axiap` component.

Table 2-3: Conditions to generate an AXI access for the `css600_axiap`

CSW.Cache + CSW.Domain are valid	AXI_DATA_WIDTH	CSW.MTE	CSW.Size written	CSW.Size read back	AXI access generated	APB error	AXI_ADDR_WIDTH
Yes	64	Read-only 0	• 8-bit • 16-bit • 32-bit • 64-bit • 64-bit	• 8-bit • 16-bit • 32-bit • 64-bit • 32-bit	• Yes, AxSIZE[2:0]=000 • Yes, AxSIZE[2:0]=001 • Yes, AxSIZE[2:0]=010 • Yes, AxSIZE[2:0]=011 • Yes, AxSIZE[2:0]=010	-	32 or 64
Yes	Otherwise assume 32	Read-only 0	• 8-bit • 16-bit • 32-bit • 64-bit • 64-bit	• 8-bit • 16-bit • 32-bit • 32-bit • 32-bit	• Yes, AxSIZE[2:0]=000 • Yes, AxSIZE[2:0]=001 • Yes, AxSIZE[2:0]=010 • Yes, AxSIZE[2:0]=010 • Yes, AxSIZE[2:0]=010	-	32 or 64

CSW.Cache + CSW.Domain are valid	AXI_DATA_ WIDTH	CSW.MTE	CSW.Size written	CSW.Size read back	AXI access generated	APB error	AXI_ADDR_ WIDTH
No	-	-	-	-	No	Yes	-

The following table shows the other conditions required to generate an AXI access for the `css600_axiap_mte` component.

Table 2-4: Conditions to generate an AXI access for the `css600_axiap_mte`

CSW.Cache + CSW.Domain are valid	AXI_DATA_ WIDTH	CSW.MTE	CSW.Size written	CSW.Size read back	AXI access generated	APB error	AXI_ADDR_ WIDTH
Yes	Ignore and assume 128	Read/write 0	• 8-bit • 16-bit • 32-bit • 64-bit • 64-bit	• 8-bit • 16-bit • 32-bit • 64-bit • 32-bit	• Yes, AxSIZE[2:0]=000 • Yes, AxSIZE[2:0]=001 • Yes, AxSIZE[2:0]=010 • Yes, AxSIZE[2:0]=011 • Yes, AxSIZE[2:0]=010	-	Ignore and assume 64
Yes	Ignore and assume 128	Read/write 1	• 8-bit • 16-bit • 32-bit • 64-bit	• 8-bit • 16-bit • 32-bit • 32-bit	No	Yes	-
Yes	Ignore and assume 128	Read/write 1	64-bit	64-bit	Yes, AxSIZE[2:0]=011	-	Ignore and assume 64
No	-	-	-	-	No	Yes	-

When no AXI access can be generated the DRW, BD, or DAR register access terminates with an APB error. The error is logged by setting TRR.ERR. See Error response handling.

2.2.3.7 AXI memory tagging

The AXI-AP with MTE supports the basic memory tagging extension. See the *AMBA® AXI Protocol Specification*.

A memory tag is n=4 bits wide (n=CFG1.TAG0SIZE=4) and exists on a N=16 byte granule of memory ($N = 2^{CFG1.TAG0GRAN} = 16$). There are M=8 memory tags, ($M = 32x/n = 8$, where x = amount of T*TR registers = 1). The memory tags are aliased on a $M \times N = 8 \times 16 = 128$ byte granule of memory.

Before the debug software initiates a memory access using the DRW, BD, or DAR registers, you must set CSW.Size, CSW.MTE, CSW.Cache, and CSW.Domain for generating an AXI access. See AXI access generation. Otherwise, the DRW, BD, or DAR register access terminates with an APB error. The error is logged by setting TRR.ERR. See Error response handling.

If the debug software initiates a memory access with CSW.MTE=0, the AXI manager performs an AXI access without asserting the MTE interface signals artagop_m, awtagop_m, and wtagupdate_m.

When the debug software initiates a:

Memory read with CSW.MTE=1

An AXI read access with memory tagging is performed on the AXI manager interface with MTE signal artagop_m=Transfer=0b01. When the AXI subordinate responds:

- Without error, it provides a 4-bit memory tag on rtag_m[3:0]. This memory tag is recorded and stored in one of the 8 tag fields in the T0TR register.
- With error, it provides no memory tag on rtag_m[3:0] signal and no tag field is recorded in T0TR. The error is logged by setting TRR.ERR. See Error response handling.

Memory write with CSW.MTE=1

An AXI write access with memory tagging is initiated on the AXI manager interface with MTE signal awtagop_m=Update=0b10. Also, a 4-bit memory tag is provided on wtag_m[3:0] with wtagupdate_m[0] asserted. The memory tag is provided from one of the 8 tag fields in the T0TR.

If the AXI subordinate responds with error, the error is logged by setting TRR.ERR. See Error response handling.

The debug software can also write to the T0TR.

2.2.3.8 Memory Encryption Contexts ID

In the MTE and non-MTE versions of the AXI Access Port (AXI-AP), support is provided for signaling the Memory Encryption Context ID (MECID) on the AXI interface. The Memory Encryption Contexts (MEC) feature was introduced in AXI issue K, and is not present in the APB or AHB APs.

The MECID support in an AXI-AP allows debug tools to provide the required encryption ID hint, when accessing memory in systems that implement memory encryption. The MECID register is writeable and the value written is output on the armecid or awmecid signals during an AXI transaction.

When the SoC-600 implements MECID, that is when legacy_tz_en = 0, the CFG.MECIDWIDTH register returns 16, which is the maximum allowed width. No other width values are used. If the system uses a MECIDWIDTH of less than 16, then unused bits of the armecid and awmecid signals are unconnected.

An access to the Realm PAS is when the following conditions are true for an active and enabled logical AP:

- (legacy_tz_en = 0) & (CSW.NSE = 1) & (CSW.Prot[1] = 1) & (CSW.RMEEN ≥ 0b01)

If legacy_tz_en is 1, the MECID feature is disabled and:

- The MECID.MECID and CFG.MECIDWIDTH register fields both return 0.
- The armecid and awmecid signals are always LOW.

2.2.4 Error response handling

CoreSight™ SoC-600 Memory Access Ports (MEM-APs) implement error response handling version 1.

Error response handling v1 is defined in the *Arm® Debug Interface Architecture Specification ADIv6.0*. Support for this error handling mechanism is indicated in the CFG.ERR register field. The three register bits CSW.ERRNPASS, CSW.ERRSTOP, and TRR.ERR define the behavior of this feature. For more information, see the relevant programmers model register descriptions.

The MEM-AP logs errors in the Transfer Response Register by setting TRR.ERR bit to 1. When set, this bit remains set until software clears it by writing 1 to it. The following types of memory access errors are logged:

Access Port Enable failure

This error is caused by an unauthenticated memory access attempt, such as:

- Any memory access when ap_en signal is LOW.
- A Secure memory access when ap_secure_en signal is LOW.

Stopped on error

This error is caused by a memory access attempt when TRR.ERR=1 and CSW.ERRSTOP=1.

AHB/APB/AXI error

An error response that is received on the AP requester interface indicating that the memory access failed. In the AXI-AP this also includes when the AXI manager is unable to start the transaction because of invalid combination of CSW. Size, CSW.MTE, CSW.Cache, and CSW.Domain. See AXI access generation and Valid combinations of AxCACHE and AxDOMAIN.

Abort

Aborted memory transfers.

Master busy

This error happens if a memory access is attempted after an abort, but while the CSW.TrInProg bit is still set.

Invalid transaction

This error only applies to the AXI-AP when a memory access is attempted with an invalid combination of CSW.Cache and CSW.Domain fields.

Internal register access errors are not logged in the TRR but are always passed on the APB completer interface. If a register write is attempted after an abort while the CSW.TrInProg bit is set, an error is generated.

The CSW.ERRNPASS register bit controls whether a memory access error is passed back to the requester. The internal register access errors are always passed back on the APB completer interface regardless of the value of this bit.

The CSW.ERRNPASS bit has the following effect on behavior:

0

Memory access errors are passed back on the APB completer interface.

1

Memory access errors are not passed back on the APB completer interface. In this case, a normal APB response is returned even for failed memory transactions.

There are two exceptions to this rule. In each case, the error is always passed on the APB completer interface, regardless of the status of the CSW.ERRNPASS bit. The exceptions are:

- If the memory transaction is aborted.
- If the error is generated by a memory access attempt, while the CSW.TrInProg bit is still set from a previously aborted access.

The APB read data for all transactions that generate an error is **UNKNOWN**.

If no previous memory access errors are logged, that is TRR.ERR=0, memory accesses are allowed, regardless of the state of CSW.ERRSTOP.

If a previous memory access error is still logged, that is TRR.ERR=1, the register field CSW.ERRSTOP controls whether to prevent memory accesses as follows:

0

New memory accesses are allowed.

1

No new memory accesses are allowed, and any new memory accesses result in an error response on the APB completer interface, provided CSW.ERRNPASS is 0. In this case, TRR.ERR remains set and the memory transfer is not initiated.

The following table summarizes this MEM-AP behavior for memory errors other than Abort and Master Busy.

Table 2-5: MEM-AP behavior for memory errors other than Abort and Master Busy

TRR.ERR	CSW.ERRNPASS	CSW.ERRSTOP	New memory access	Completer error	Error logged
0	0	x	Allowed if authenticated by the Access Port Enable interface, otherwise blocked	Passed	Yes
0	1	x	Allowed if authenticated by the Access Port Enable interface, otherwise blocked	Not passed	Yes
1	0	0	Allowed if authenticated by the Access Port Enable interface, otherwise blocked	Passed	Yes
1	1	0	Allowed if authenticated by the Access Port Enable interface, otherwise blocked	Not passed	Yes
1	0	1	Blocked	Passed	Yes
1	1	1	Blocked	Not passed	Yes

The twin logical APs implement error handling independently, so the errors that one AP receives or generates do not affect the other AP.

Memory errors, other than Abort and Master-Busy, are maskable errors. That is, they can be masked from appearing on an APB completer interface by setting the CSW.ERRNPASS bit. It is possible for a single memory access to cause multiple error sources to generate errors at the same time. For example, a memory access can trigger a stop-on-error and an authentication failure.

If an error is masked, an error response is passed on the APB completer interface, even if CSW.ERRNPASS is 1, and if at least one of the sources of error is non-maskable (Abort or Master-Busy). If all the triggered error sources are maskable, the error is passed only if CSW.ERRNPASS is 0.

If the Access Port Enable interface signals change while a memory transfer is in progress, the MEM-AP still completes the ongoing transfer normally. The new Access Port Enable interface values then take effect from the next transaction. If the MEM-AP receives a memory access request in the Q_STOPPED state, then it samples the authentication signal values only after entering the first cycle of Q_RUN state. The sampled value controls whether to allow or block the pending APB transfer.

2.2.5 Root and realm transactions

The SoC-600 Memory Access Ports (MEM-APs) that support the Realm Management Extension (RME) are APB-AP and AXI-AP.

2.2.5.1 Realm Management Extension

Realm Management Extension (RME) is the hardware component of the Arm Confidential Compute Architecture (Arm CCA) which also includes software elements.

RME dynamically transfers resources and memory to a new protected address space that higher privileged software or TrustZone® firmware cannot access. Because of this address space, Arm CCA constructs protected execution environments called realms. See the *Learn the architecture - Realm Management Extension*.

Realms allow a lower-privileged software, like an application or a Virtual Machine (VM), to protect its content. Realms also prevent execution from attacks using software that runs at higher privilege levels, like an OS or a hypervisor. Higher-privileged software is still responsible for allocating and managing the resources that a realm uses. However, this higher-privileged software cannot access the contents of the realm or affect its execution flow.

RME can also dynamically transfer memory to a protected address space for realms. With RME, the memory available to TrustZone software entities can be varied dynamically.

The two logical APs are functionally separate from each other, and only share the APB requester or AXI manager interface. Because of this separation, there are no programmer-visible changes when the AP is configured with separate enables.

2.2.5.2 Authentication signals and registers in a Memory Access Port

Access to the downstream memory system is determined by various input signals to the SoC-600 Memory Access Port (MEM-AP) and the type of access it receives.

The CFG, AUTHSTATUS, and CSW registers are used to indicate the capability provided and set the requested physical address space (PAS). Therefore, these registers control whether the MEM-AP allows the requested transaction to occur or ignores it.

There are some differences between the specific capabilities of different MEM-APs. See the detailed programmers model for the AP that is being used.

CFG register

The CFG.RME bit indicates if the Realm Management Extension (RME) feature is implemented (1) or not implemented (0). This is normally fixed for a given implementation. CFG.RME is present only in the APB and AXI Access Ports. In the AHB-AP, this bit is **RES0**.

AUTHSTATUS register

The Authentication Status register contains a read-only view of the current capabilities of the MEM-AP. The fields of the AUTHSTATUS register report the capability relating to:

- Non-secure debug enabled state
- Secure debug enabled state
- Realm debug enabled state
- Root debug enabled state

The AUTHSTATUS register of each logical AP reflects the values of the authentication interface signals applicable to that logical AP.

There is no distinction between invasive debug and non-invasive debug in a MEM-AP. Therefore, for any pair of related capabilities, the MEM-AP reports the same value for both invasive and non-invasive debug. For example, if Secure invasive debug is enabled, then Secure non-invasive debug is also enabled.

CSW register

The following CSW bits report the capability provided, based on input signals.

CSW.DeviceEn

Read-only status indicating that at least the lowest level of debug capability is enabled (ap_en input signal is HIGH).

CSW.SDeviceEn

Read-only status indicating Secure device enable. Set if the ap_en AND ap_secure_en signals are both HIGH. Secure and Root transfers are allowed only if these bits are set.

CSW.NSE, CSW.Prot[1], CSW.HNONSEC

Read and write control bits that select between Non-secure, Secure, Root, and Realm physical address spaces (PAS). See Writeable register settings for PAS accesses for more information.

MEM-AP authentication behavior

The following tables describe the details of the register contents and show when a downstream transaction is allowed to occur and to which PAS.

For information about how the register bits are set, see the corresponding register value calculations in the Access Port Authentication interface section of the *Arm® CoreSight™ System-on-Chip SoC-600 Configuration and Integration Manual*.

Table 2-6: Register content to capability indication for non-RME configurations

CFG.RME	CSW.DeviceEn	CSW.SDeviceEn	CSW.RMEEN	Non-secure permitted	Secure permitted	Realm permitted	Root permitted
0	0	0	0b00	No	No	No	No
0	1	0	0b00	Yes	No	No	No
0	1	1	0b00	Yes	Yes	No	No

Table 2-7: Register content to capability indication for RME configurations

CFG.RME	CSW.DeviceEn	CSW.SDeviceEn	CSW.RMEEN	Non-secure permitted	Secure permitted	Realm permitted	Root permitted
1	0	0	0b00	No	No	No	No
1	1	0	0b00	Yes	No	No	No
1	1	0	0b01	Yes	No	Yes	No
1	1	0	0b11	Yes	No	Yes	No
1	1	1	0b00	Yes	Yes	No	No
1	1	1	0b01	Yes	Yes	Yes	No
1	1	1	0b11	Yes	Yes	Yes	Yes

Writeable register settings for PAS accesses

The following tables assume that the conditions for allowing a given transaction are present, according to the previous tables. If the required conditions are not present, then the requested downstream transaction does not occur.

The APB and AXI Access ports use the CSW.Prot[1] bit to control the Secure or Non-secure selection, whereas the AHB-AP uses the CSW.HNONSEC bit.

Only the APB and AXI Access ports support the Realm Management Extension, so they implement the CSW.NSE bit. In the AHB-AP, the CSW.NSE bit is **RES0**.

Table 2-8: APB-AP and AXI-AP PAS selection

Requested transaction type	CSW.Prot[1]	CSW.NSE
Non-secure	1	0

Requested transaction type	CSW.Prot[1]	CSW.NSE
Secure	0	0
Realm	1	1
Root	0	1

Table 2-9: AHB-AP PAS selection

Requested transaction type	CSW.HNONSEC
Non-secure	1
Secure	0

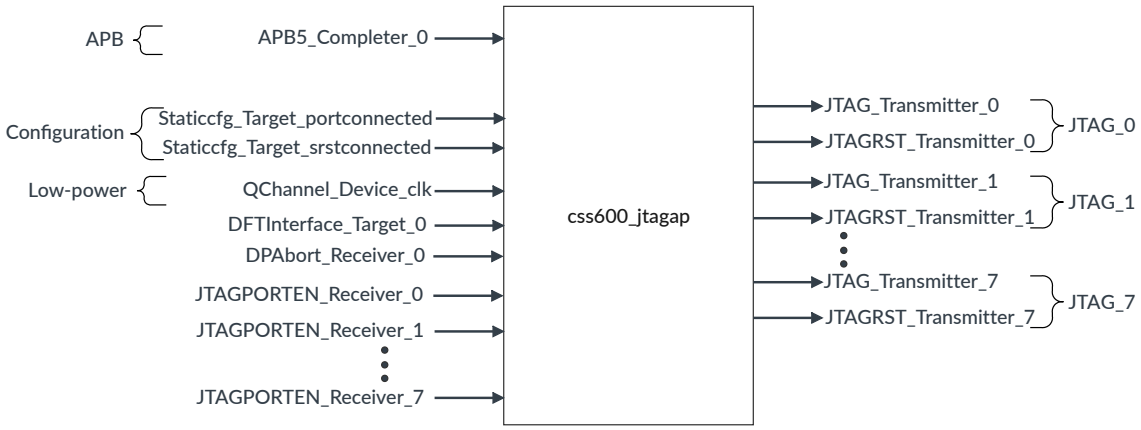
2.3 JTAG Access Port

The `css600_jtagap` provides JTAG access to on-chip components, operating as a JTAG transmitter port to drive JTAG chains throughout a SoC.

The JTAG command protocol is byte-oriented, with a word wrapper on the read and write ports to yield acceptable performance from the 32-bit internal data bus in the DAP. Daisy chaining is avoided by using a port multiplexer. These two features prevent slower cores from impeding faster cores.

The following figure shows the external connections on the JTAG Access Port.

Figure 2-6: css600_jtagap logical connections



For more information, see the *Arm® Debug Interface Architecture Specification ADIv6.0*.

2.4 Access Port v1 adapter

Use the `css600_apv1adapter` to connect a legacy Access Port (AP) with a DAP Internal (DAPBus) completer interface into a CoreSight™ architecture v3 system.

The `css600_apv1adapter`:

- Maps the legacy AP registers into the *Arm® CoreSight™ Architecture Specification v3.0* APB5 memory map
- Provides ID registers that allow a debugger to identify the combination as a mapped legacy Access Port
- Provides integration registers that allow a debugger to check connectivity of the dp_abort signal

The following figure shows the external connections on the Access Port v1 Adapter.

Figure 2-7: css600_apv1adapter logical connections

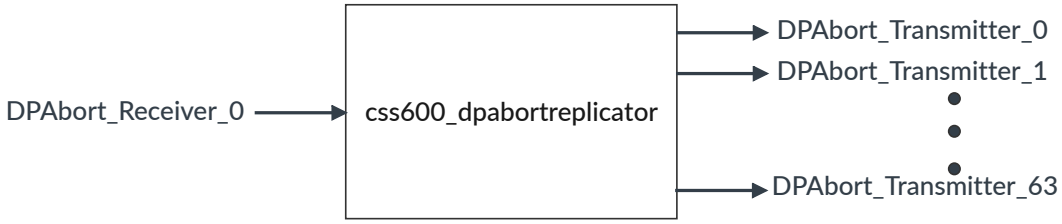
2.5 DP Abort replicator

The `css600_dpabortreplicator` is an IP-XACT phantom component that supports stitching in an IP-XACT tooling product. There is no Verilog module for `css600_dpabortreplicator`.

Use the `css600_dpabortreplicator` to connect a single DP Abort transmitter interface to multiple DP Abort receiver interfaces. You must connect the dp_abort output signal from the Debug Port to every Access Port that appears in the Debug Port memory space.

The following figure shows the external connections on the DP Abort Replicator.

Figure 2-8: css600_dpabortreplicator logical connections



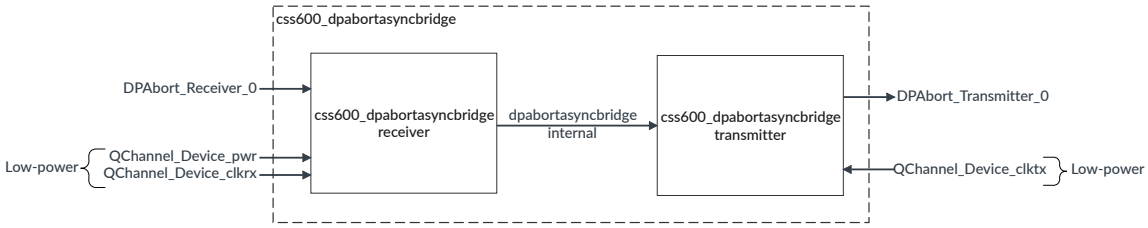
2.6 DP Abort asynchronous bridge

The `css600_dpabortasyncbridge` is a wrapper component that instantiates a pulse asynchronous bridge.

The bridge transfers the dp_abort signal across a clock or power domain boundary. The dp_abort signal is a pulse event that unlocks a deadlocked transaction on a DP to AP interconnection.

The following figure shows the external connections on the DP Abort asynchronous bridge.

Figure 2-9: css600_dpabortasyncbridge logical connections



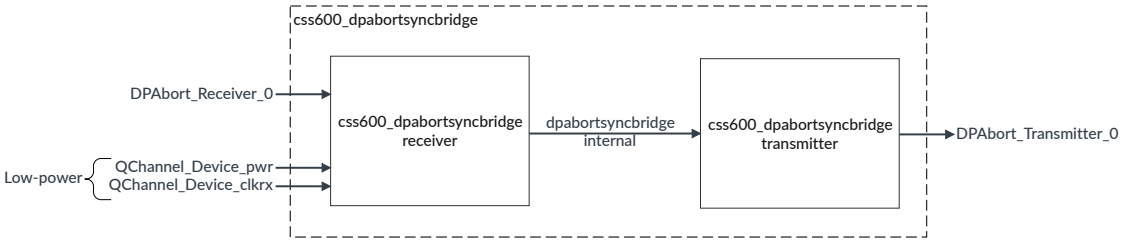
2.7 DP Abort synchronous bridge

The `css600_dpabortsyncbridge` is a wrapper component that instantiates a pulse synchronous bridge.

The bridge transfers the dp_abort signal across a power domain boundary. The dp_abort signal is a pulse event that unlocks a deadlocked transaction on a DP to AP interconnection.

The following figure shows the external connections on the DP Abort synchronous bridge.

Figure 2-10: css600_dpabortsyncbridge logical connections



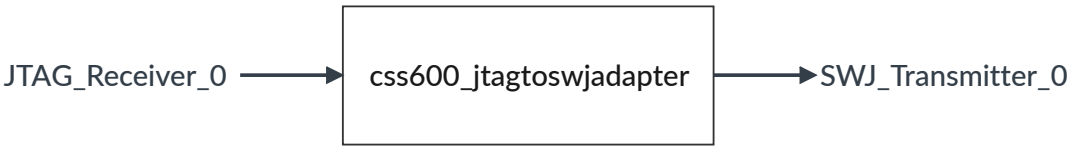
2.8 JTAG to SWJ adapter

The `css600_jtagtoswjadapter` is an IP-XACT phantom component that supports stitching in an IP-XACT tooling product. There is no Verilog module for `css600_jtagtoswjadapter`.

Use the `css600_jtagtoswjadapter` to connect a JTAG transmitter interface to a Serial Wire/JTAG (SWJ) receiver interface. This might be necessary when connecting a Debug Port to the `css600_jtagap`.

The following figure shows the external connections on the JTAG to SWJ adapter.

Figure 2-11: css600_jtagtoswjadapter logical connections



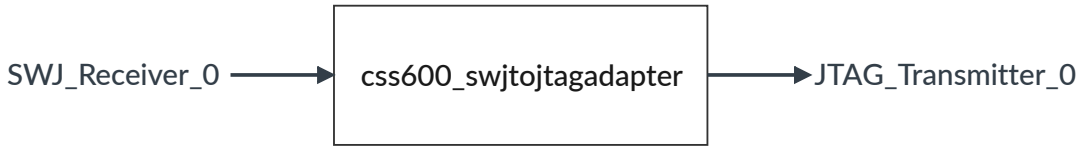
2.9 SWJ to JTAG adapter

The `css600_swjtojtagadapter` supports stitching in an IP-XACT tooling product.

Use the `css600_swjtojtagadapter` to connect a Serial Wire/JTAG (SWJ) transmitter interface to a JTAG receiver interface.

The following figure shows the external connections on the SWJ to JTAG adapter.

Figure 2-12: css600_swjtojtagadapter logical connections



2.10 SWJ interconnect

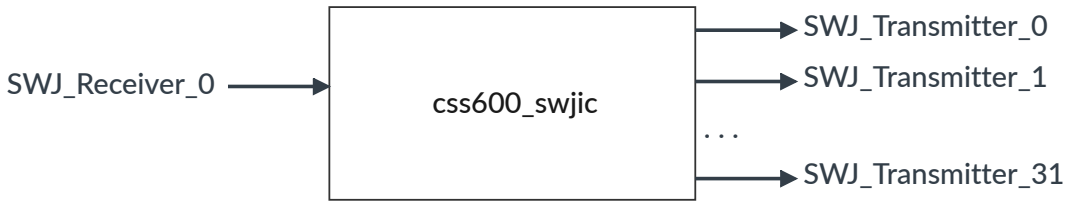
The `css600_swjic` is a Serial Wire and JTAG interconnect that enables you to connect multiple SWJ receiver components, for example Debug Ports, to a single SWJ transmitter.

Use the `css600_swjic` to:

- Daisy-chain multiple JTAG Debug Ports
- Combine the data and control signals from multiple Serial Wire multi-drop Debug Ports

The following figure shows the external connections on the SWJIC.

Figure 2-13: css600_swjic logical connections



Note

Creating long JTAG scan chains can create performance issues. We recommend that you avoid creating long daisy-chains if possible.

3. APB infrastructure components functional description

The SoC-600 APB infrastructure components provide an APB interconnect and supporting components.

3.1 APB interconnect

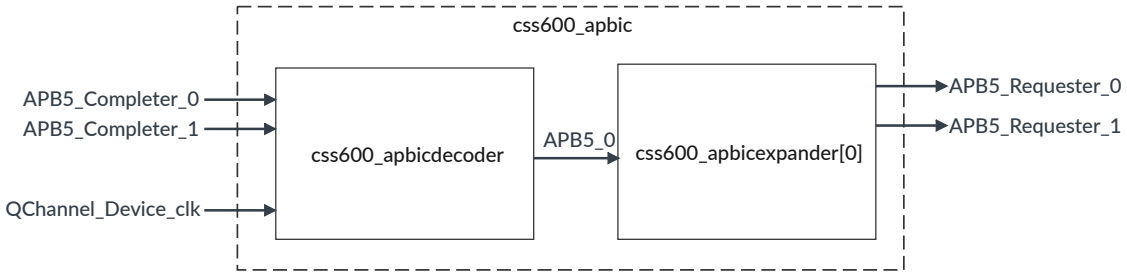
The `css600_apbic` provides connections between APB5 requesters and APB5 completers anywhere in a CoreSight™ system.

APB5 requesters can be debug ports, APB Access Ports, or other APB requesters from a compute subsystem. It is a two-part meta-component that has the following features:

- A single clock domain.
- A decoder component configurable for up to four completer interfaces and up to 64 requester interfaces.
- Physical grouping of decoded requester interfaces using one or more configurable expander components.
- The option to insert APB asynchronous or synchronous bridges between decoder and expander instances to cross power and clock domain boundaries.
- Configurable APB address widths to suit addressable ranges.
- A Q-Channel LPI for high-level clock management.

The following figure shows the external connections on the APB interconnect.

Figure 3-1: css600_apbic logical connections



3.1.1 Arbitration

The internal arbiter arbitrates between competing completer interfaces for access to the debug APB.

When a completer interface raises a request, the arbiter gives the highest priority to the completer interface with the lowest instance suffix. For example, Completer Interface 0 > Completer Interface 1 > Completer Interface 2 > Completer Interface 3. The order in which the completer interfaces raised their requests relative to each other is not used in arbitration.

The arbitration is re-evaluated after every access.

3.1.2 Error response

The SoC-600 APB interconnect returns an error on its completer interface under certain conditions.

An error response is returned when either:

- The targeted APB completer returns an error response.
- A completer interface accesses an address that does not decode to any connected APB completer.

3.2 APB ROM table

The `css600_apbrom` module is a ROM table with an APB5 completer interface.

The `css600_apbrom_gpr` is a ROM table that includes the Granular Power Requestor (GPR) function.

The ROM table is implemented according to the *Arm® CoreSight™ Architecture Specification v3.0* and the *Arm® Debug Interface Architecture Specification ADIv6.0*.

Use the `css600_apbrom` or `css600_apbrom_gpr` to:

- Identify part of your system or subsystem.
- Indicate the locations of other CoreSight™ components in the same address space to an external debugger.
- Request power or reset to be supplied to components in the debug subsystem or the wider system (`css600_apbrom_gpr only`).

The `css600_apbrom` and `css600_apbrom_gpr`, by default, support up to 512 32-bit component entries, which you set with configuration parameters. You can configure `css600_apbrom` and `css600_apbrom_gpr` to support 256 64-bit entries. This enables the APB ROM table to point to components located in larger address spaces.

The `css600_apbrom` and `css600_apbrom_gpr` support dynamic control of the ROM table IDs, and optionally, the presence of each entry, using configuration input signals. These features make the ROM table suitable for use in configurable and hardened subsystems.

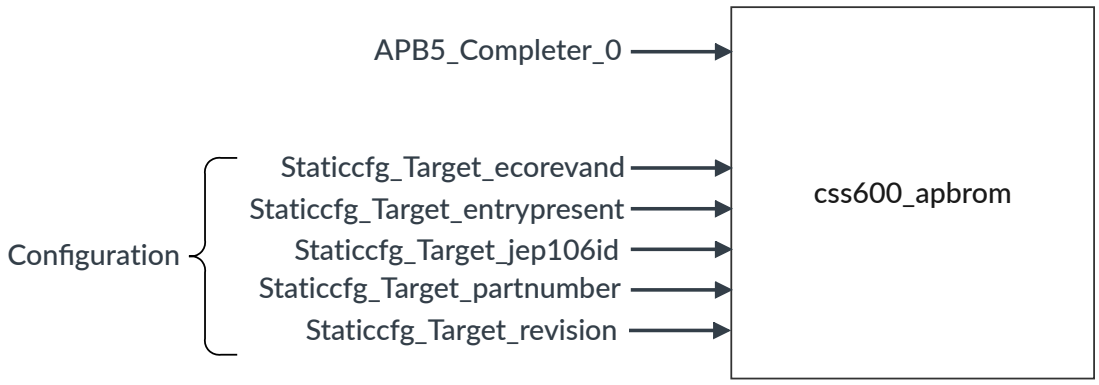
The `css600_apbrom_gpr` is able to request power or reset to individual components or parts of a system through a power or reset controller that is implemented outside the CoreSight subsystem. Power request interface numbers are normally aligned to power domain IDs configured into the ROM table.

The GPR configuration provides the following additional features:

- An authentication interface to control access to power and reset control features.
- A configurable number, up to 32, of debug power request interfaces, comprising a cdbgpwrupreq and cdbgpwrupack pair of signals.
- A configurable number, up to 32, of system power request interfaces, comprising a csyspwrupreq and csyspwrupack pair of signals.
- A debug reset request interface, comprising a cdbgrstreq and cdbgrstack pair of signals.
- A system reset request interface, comprising a csysrstreq and csysrstack pair of signals.

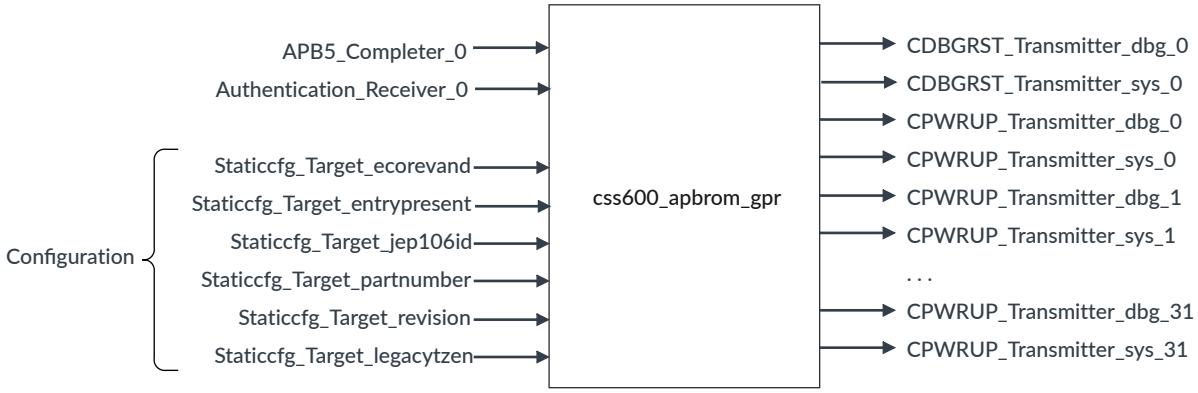
The following figure shows the external connections on the APB ROM table.

Figure 3-2: css600_apbrom logical connections



The following figure shows the external connections on the APB ROM table with GPR.

Figure 3-3: css600_apbrom_gpr logical connections



3.3 APB asynchronous bridge

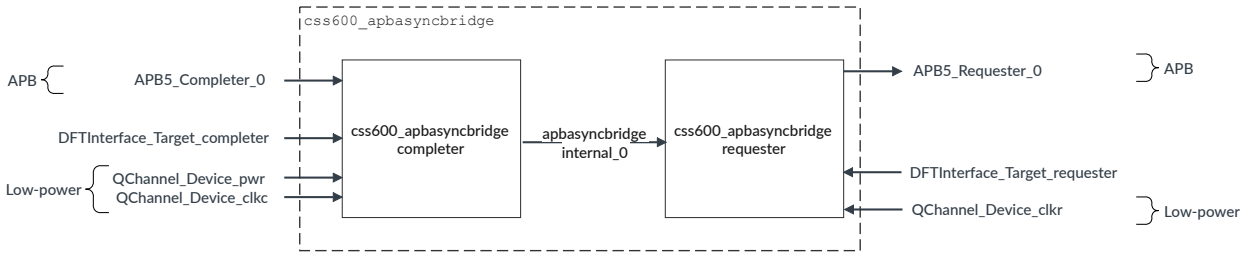
The `css600_apbasyncbridge` component is used where an AMBA® APB5 bus is required to cross a clock or power domain boundary.

The SoC-600 APB asynchronous bridge provides the following features:

- Two independent clock domains with any phase or frequency alignment.
- Two independent power domains, either of which can be switched relative to the other.
- Three Q-Channel LPIs for completer side clock, requester side clock, and power switching management.
- A two-part meta-component with separate completer and requester side components.
- Configurable APB address width.
- Configurable 2-deep or 3-deep synchronizers.

The following figure shows the external connections on the APB asynchronous bridge.

Figure 3-4: css600_apbasyncbridge logical connections



3.4 APB synchronous bridge

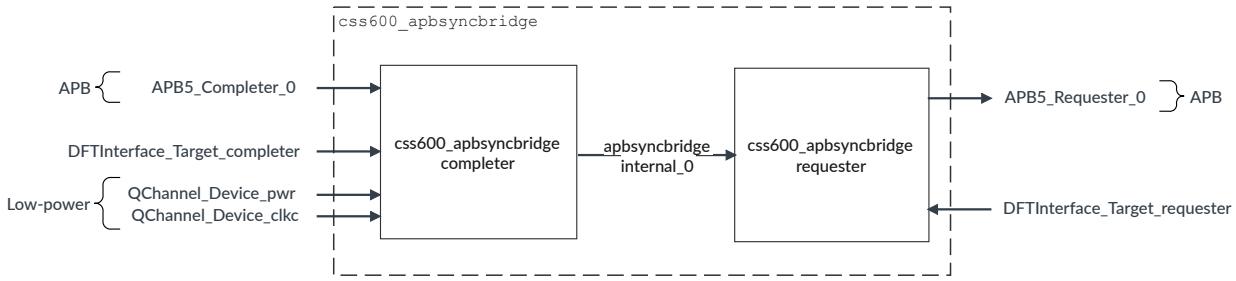
The `css600_apbsyncbridge` is used where an AMBA® APB5 bus is required to cross a clock domain boundary between two synchronous clocks.

The APB asynchronous bridge provides the following features:

- Two synchronous clock domains with any frequency difference. The clocks must be skew balanced and from a common source, so that they are high-level-gated by a common control point.
- Two Q-Channel LPIs for clock and power switching management.
- Two-part meta-component with separate completer and requester side components.
- Configurable APB address width.
- Configurable 2-deep or 3-deep synchronizers for Q-Channel inputs.

The following figure shows the external connections on the APB synchronous bridge.

Figure 3-5: css600_apbsyncbridge logical connections



3.5 APB PADDRDBG31 adapter

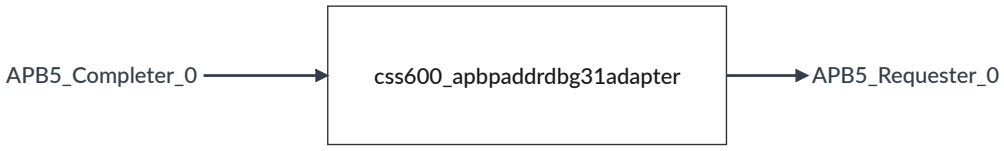
The `css600_apbpaddrdbg31adapter` module enables you to integrate a CoreSight™ Architecture v2.0 component or subsystem into a CoreSight debug and trace subsystem (CSSYS).

Use the `css600_apbpaddrdbg31adapter` to integrate legacy components that have a dedicated paddrdbg31 signal into the memory map of the CoreSight SoC-600 CSSYS. The `css600_apbpaddrdbg31adapter` component:

- Replaces the 2GB split at `0x80000000` with a user-defined split
- Maps the two views of the component to consecutive regions of the memory map
- Maps the external debugger view to the lower region
- Maps the self-hosted view to the upper region

The following figure shows the external connections on the APB PADDRDBG31 adapter.

Figure 3-6: css600_apbpaddrdbg31adapter logical connections



4. AMBA Trace Bus infrastructure components functional description

The AMBA® Trace Bus (ATB) infrastructure components move trace stream data from trace sources to one or more suitable destinations.

4.1 ATB upsizer

The SoC-600 `css600_atbupsizer` component enables you to increase the data width of an AMBA® Trace Bus (ATB).

Use the `css600_atbupsizer` when you connect an ATB transmitter interface to a wider ATB receiver interface.

The following figure shows the external connections on the ATB upsizer.

Figure 4-1: css600_atbupsizer logical connections

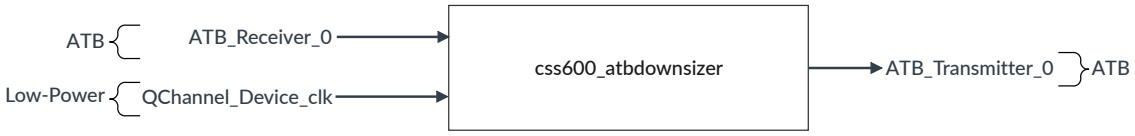
4.2 ATB downsizer

The SoC-600 `css600_atbdownsizer` component enables you to reduce the data width of an AMBA® Trace Bus (ATB).

Use the `css600_atbdownsizer` when you must connect an ATB transmitter interface to a narrower ATB receiver interface.

The following figure shows the external connections on the ATB downsizer.

Figure 4-2: css600_atbdownsizer logical connections



4.3 ATB funnel

The SoC-600 `css600_atbfunnel` component is used when more than one trace source must be merged into a single trace stream.

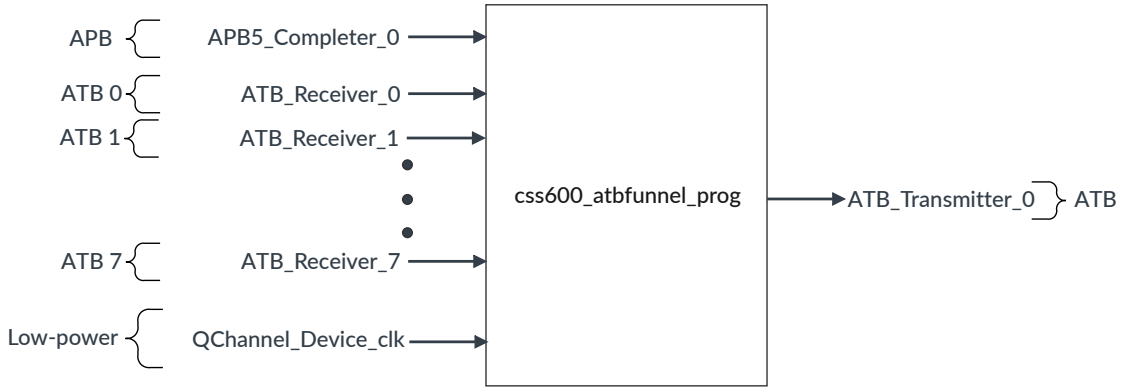
The funnel is configurable for the number of receiver interfaces, from 2-8, and comes in programmable or non-programmable configurations. The programmers model section describes the register map of the programmable version.

The programmable configuration allows the following features:

- Independent enable control for each receiver port.
- Independent priority setting for each receiver port, so that higher priority ports are serviced ahead of lower priority ports.
- Programmable hold time to reduce input switching that is based on trace ID value.
- Registers that allow integration testing of the trace network.

The following figure shows the external connections on the programmable AMBA® Trace Bus (ATB) funnel.

Figure 4-3: css600_atbfunnel_prog logical connections

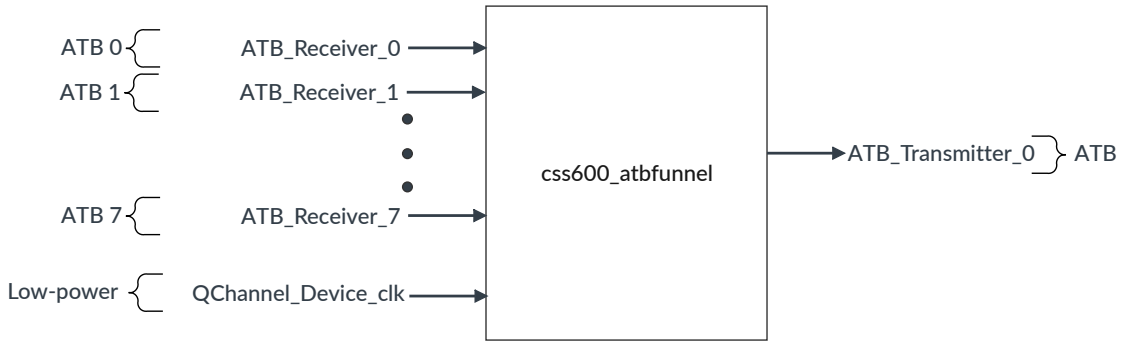


The non-programmable configuration has the following features:

- All receiver ports are enabled.
- All receiver ports have equal priority.
- All receiver ports have a hold time of four transactions.

The following figure shows the external connections on the non-programmable ATB funnel.

Figure 4-4: css600_atbfunnel logical connections



4.4 ATB replicator

The SoC-600 `css600_atbreplicator` component splits a single trace stream into two trace streams for systems that have more than one trace sink component.

An optional programmable configuration is available that provides the following features:

- Filtering of trace IDs to allow some IDs to go to transmitter port 0 and some to transmitter port 1.
- Registers that allow integration testing of the trace network.

The following figure shows the external connections on the programmable AMBA® Trace Bus (ATB) replicator.

Figure 4-5: css600_atbreplicator_prog logical connections

In the non-programmable configuration, no ATB ID filtering is applied to either transmitter.

The following figure shows the external connections on the non-programmable ATB replicator.

Figure 4-6: css600_atbreplicator logical connections

4.5 ATB trace buffer

The SoC-600 `css600_atbbuffer` component is used in situations where some local smoothing of trace bandwidth is required in a trace network.

The AMBA® Trace Bus (ATB) trace buffer has the following features:

- Configurable trace data width up to 128 bits
- Configurable buffer depth up to 256 entries
- Configurable threshold for buffer fill level before starting to empty

The following figure shows the external connections on the ATB trace buffer.

Figure 4-7: css600_atbbuffer logical connections

4.6 ATB asynchronous bridge

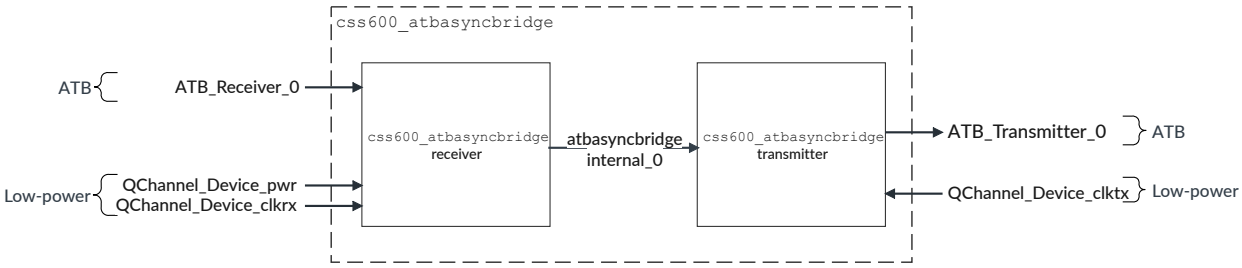
The SoC-600 `css600_atbasyncbridge` component transports the AMBA® Trace Bus (ATB) across a clock or power domain boundary.

The ATB asynchronous bridge provides the following features:

- Two independent clock domains with any phase or frequency alignment.
- Two independent power domains, either of which can be switched relative to the other.
- Three Q-Channel LPIs for receiver side clock, transmitter side clock, and power switching management.
- Two-part meta-component with separate receiver and transmitter side components.
- Configurable ATB data width.
- Configurable for 2-stage or 3-stage synchronizers.
- Automatically manages upstream flush of trace data before power down.

The following figure shows the external connections on the ATB asynchronous bridge.

Figure 4-8: css600_atbasyncbridge logical connections



4.7 ATB synchronous bridge

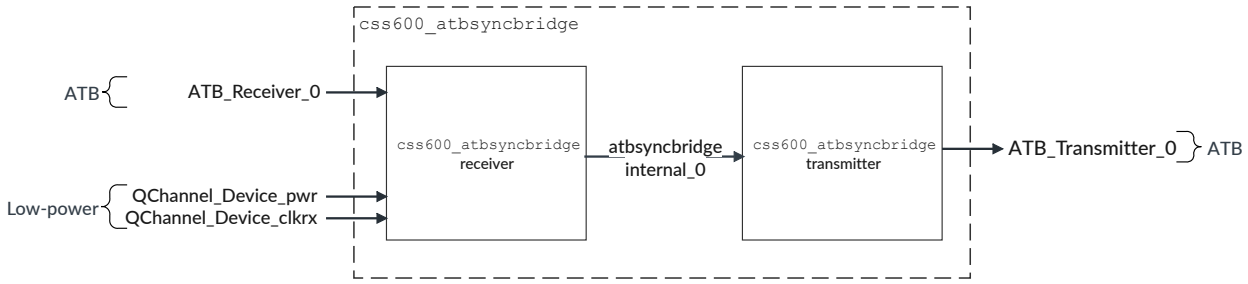
The SoC-600 `css600_atbsyncbridge` component transports the AMBA® Trace Bus (ATB) across a clock domain boundary.

The ATB synchronous bridge provides the following features:

- Two synchronous clock domains with any frequency difference. The clocks must be skew balanced and from a common source so that they are high-level-gated by a common control point.
- Two Q-Channel LPIs for clock and power switching management.
- Two-part meta-component with separate receiver and transmitter side components.
- Configurable ATB data width.
- Configurable for 2-stage or 3-stage synchronizers.
- Automatically manages upstream flush of trace data before power down.

The following figure shows the external connections on the ATB synchronous bridge.

Figure 4-9: css600_atbsyncbridge logical connections



4.8 Trace Memory Controller

The SoC-600 Trace Memory Controller (TMC) captures trace data into local or system memory, or streams it to a High Speed Serial Trace port.

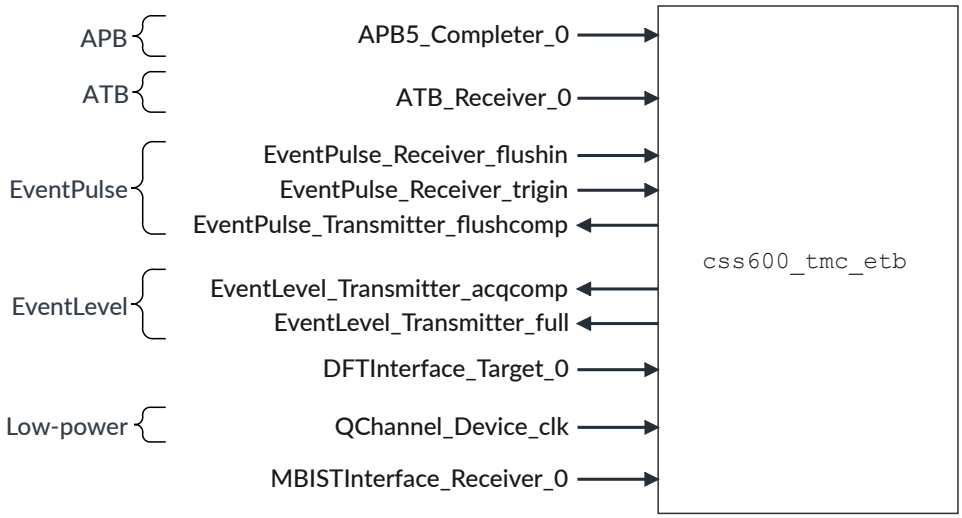
An off-chip external debugger or on-chip self-hosted debug software can read the trace.

The Trace Memory Controller is made up of four TMC components:

`css600_tmc_etb` Embedded Trace Buffer (ETB)

Enables trace to be stored in a dedicated SRAM within the TMC module that is used as a circular buffer. The following figure shows the external connections of the TMC ETB component.

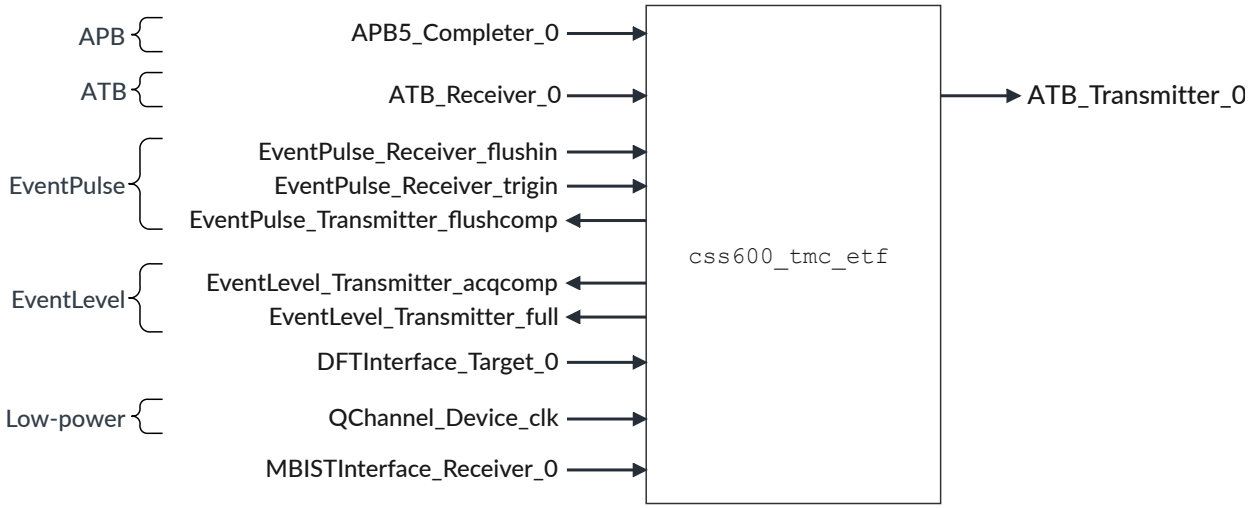
Figure 4-10: css600_tmc_etb logical connections



`css600_tmc_etf` Embedded Trace FIFO (ETF)

Enables trace to be stored in a dedicated SRAM within the TMC module that is used either as a circular buffer or as a FIFO. The functionality of the ETF component is a superset of the functionality of the ETB component. In a CoreSight™ system, the ETF can be inserted anywhere on the trace bus and used as a FIFO to smooth out a bursty trace. The following figure shows the external connections.

Figure 4-11: css600_tmc_etf logical connections



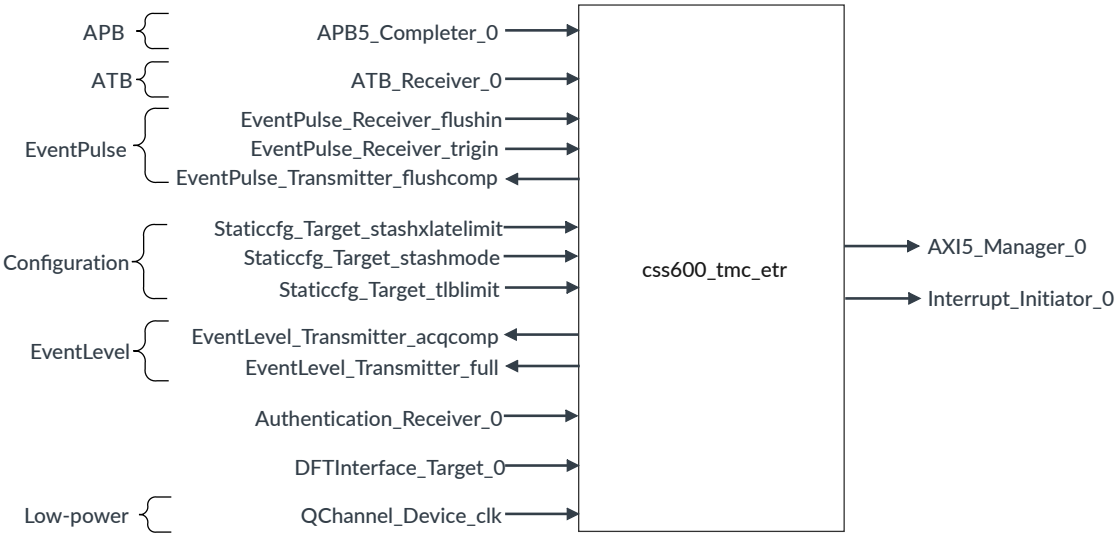
`css600_tmc_etr` Embedded Trace Router (ETR)

Enables trace to be routed over an AXI interface to the system memory or to any other AXI subordinate. You can configure the ETR for AXI address widths of 32, 40, 44, 48, 52, and 64 bits.

Note

The `css600_tmc_etr` Embedded Trace Router (ETR) writes to a single address range. If you want to write to multiple addresses, please use the CoreSight Address Translation Unit (CATU) component. See CoreSight address Translation Unit.

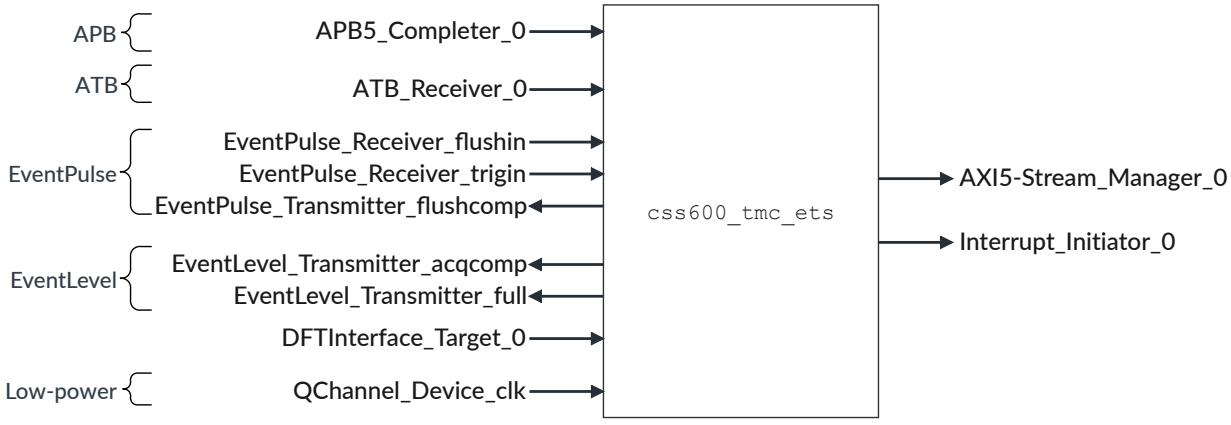
Figure 4-12: css600_tmc_etr logical connections



`css600_tmc_ets` Embedded Trace Streamer (ETS)

Enables trace to be routed over an AXI5-Stream interface to a streaming device such as an HSSTP link layer, either directly or through an AXI5-Stream interconnect. ETS retains a subset of the ETR feature set, and removes those features not required for the streaming application, and provides a lower gate count than an ETR uses for streaming. The following figure shows the external connections on the TMC ETS component.

Figure 4-13: css600_tmc_ets logical connections



The TMC can be programmed to capture trace in different modes:

- Circular Buffer mode, see Circular Buffer mode
- Software FIFO mode 1, see Software FIFO mode 1
- Software FIFO mode 2, see Software FIFO mode 2
- Hardware FIFO mode, see Hardware FIFO mode

4.8.1 TMC register access dependencies

Not all TMC registers can be read and written under the same conditions.

4.8.1.1 Writes to TMC registers

You can only write to TMC registers under specific conditions.

The following table shows the conditions that are necessary to write to each TMC register. Writing to the TMC under conditions other than those listed results in **UNPREDICTABLE** behavior. An x indicates that any value is permitted.

Table 4-1: Conditions for write accesses to TMC registers

Register	Name	Offset	CTL.TraceCaptEn	STS.TMCReady	MODE	ITCTRL.IME
RSZ (ETR)	RAM Size register	0x004	0	1	x	0
STS.MemErr (ETR)	Status Register	0x00C	x	1	x	0
STS.Full (ETR, ETS)	Status Register	0x00C	0	1	x	0
RRD	RAM Read Data Register	0x010	Read-only	Read-only	Read-only	Read-only
RRP (ETB,ETF,ETR)	RAM Read Pointer Register	0x014	0	1	x	0
RWP (ETB,ETF,ETR)	RAM Write Pointer Register	0x018	0	1	x	0
TRG	Trigger Counter Register	0x01C	0	1	x	0
CTL	Control Register	0x020	x	x	x	0
RWD (ETB,ETF,ETR)	RAM Write Data Register	0x024	0	1	x	0
MODE	Mode Register	0x028	0	1	x	0
LBUFLEVEL (ETB,ETF,ETR)	Latched Buffer Fill Level	0x02C	Read-only	Read-only	Read-only	Read-only
CBUFLEVEL (ETB,ETF,ETR)	Current Buffer Fill Level	0x030	Read-only	Read-only	Read-only	Read-only
BUFWM (ETB,ETF,ETR)	Buffer Level Water Mark	0x034	0	1	x	0
RRPHI (ETR)	RAM Read Pointer High Register	0x038	0	1	x	0
RWPHI (ETR)	RAM Write Pointer High Register	0x03C	0	1	x	0
AXICTL (ETR)	AXI Control Register	0x110	0	1	x	0
DBALO (ETR)	Data Buffer Address Low Register	0x118	0	1	x	0

Register	Name	Offset	CTL.TraceCaptEn	STS.TMCReady	MODE	ITCTRL.IME
DBAHI (ETR)	Data Buffer Address High Register	0x11C	0	1	x	0
RURP (ETR)	RAM Update Read Pointer Register	0x120	1	x	SWF2	0
FFSR	Formatter and Flush Status Register	0x300	Read-only	Read-only	Read-only	Read-only
FFCR.EmbedFlush	Formatter and Flush Control Register	0x304	x	x	x	0
FFCR.DrainBuffer (ETF)	Formatter and Flush Control Register	0x304	1	1	CB	0
FFCR.StopOnTrigEvt	Formatter and Flush Control Register	0x304	1	x	CB	0
FFCR.StopOnTrigEvt	Formatter and Flush Control Register	0x304	0	x	x	0
FFCR.StopOnFl	Formatter and Flush Control Register	0x304	x	x	x	0
FFCR.TrigOnFl	Formatter and Flush Control Register	0x304	x	x	x	0
FFCR.TrigOnTrigEvt	Formatter and Flush Control Register	0x304	1	x	CB	0
FFCR.TrigOnTrigEvt	Formatter and Flush Control Register	0x304	0	x	x	0
FFCR.TrigOnTrigIn	Formatter and Flush Control Register	0x304	x	x	x	0
FFCR.FlushMan	Formatter and Flush Control Register	0x304	x	x	x	0
FFCR.FOnTrigEvt	Formatter and Flush Control Register	0x304	1	x	CB	0
FFCR.FOnTrigEvt	Formatter and Flush Control Register	0x304	0	x	x	0
FFCR.FOnFlIn	Formatter and Flush Control Register	0x304	x	x	x	0
FFCR.EnTI	Formatter and Flush Control Register	0x304	0	1	x	0
FFCR.EnFt	Formatter and Flush Control Register	0x304	0	1	x	0
PSCR.PSCount	Periodic Synchronization Counter Register	0x308	0	1	x	0
PSCR.EmbedSync (ETR,ETS)	Periodic Synchronization Counter Register	0x308	0	1	x	0
AXICTL1 (ETR)	AXI Control Register 1	0xED0	0	1	0	0
ITATBMDATA0 (ETF)	Integration Test ATB Transmitter Data 0 Register	0xED0	0	1	x	1
ITATBMCTR2 (ETF)	Integration Test ATB Transmitter Interface Control 2 Register	0xED4	Read-only	Read-only	Read-only	Read-only
ITATBMCTR1 (ETF)	Integration Test ATB Transmitter Interface Control 1 Register	0xED8	0	1	x	1

Register	Name	Offset	CTL.TraceCaptEn	STS.TMCReady	MODE	ITCTRL.IME
ITATBMCTR0 (ETF)	Integration Test ATB Transmitter Interface Control 0 Register	0xEDC	0	1	x	1
ITEVTINTR.ACQCOMP ITEVTINTR.FULL ITEVTINTR.FLUSHCOMP	Integration Test Event & Interrupt Status Register	0xEE0	0	1	x	1
ITEVTINTR.BUFINTR (ETR,ETS)	Integration Test Event & Interrupt Status Register	0xEE0	0	1	x	1
ITTRFLIN	Integration Test Trigger In and Flush In Register	0xEE8	Read-only	Read-only	Read-only	Read-only
ITATBDATA0	Integration Test ATB Data 0 Register	0xEEC	Read-only	Read-only	Read-only	Read-only
ITATBCTR2	Integration Test ATB Control 2 Register	0xEF0	0	1	x	1
ITATBCTR1	Integration Test ATB Control 1 Register	0xEF4	Read-only	Read-only	Read-only	Read-only
ITATBCTR0	Integration Test ATB Control 0 Register	0xEF8	Read-only	Read-only	Read-only	Read-only
ITCTRL	Integration Mode Control Register	0xF00	0	1	x	x
CLAIMSET	Claim Tag Set Register	0xFA0	x	x	x	x
CLAIMCLR	Claim Tag Clear Register	0xFA4	x	x	x	x
AUTHSTATUS	Authentication Status Register	0xFB8	Read-only	Read-only	Read-only	Read-only
DEVARCH	Device Architecture Register	0xFBC	Read-only	Read-only	Read-only	Read-only
DEVID1	Device Configuration Register 1	0xFC4	Read-only	Read-only	Read-only	Read-only
DEVID	Device Configuration Register	0xFC8	Read-only	Read-only	Read-only	Read-only
DEVTYPE	Device Type Identifier Register	0xFCC	Read-only	Read-only	Read-only	Read-only
PIDR4-7	Peripheral ID Registers 4-7	0xFD0-0xFDC	Read-only	Read-only	Read-only	Read-only
PIDR0-3	Peripheral ID Registers 0-3	0xFE0-0xFEC	Read-only	Read-only	Read-only	Read-only
CIDR0-3	Component ID Registers 0-3	0xFF0-0xFFC	Read-only	Read-only	Read-only	Read-only

4.8.1.2 Reads from TMC registers

You can only read from TMC registers under specific conditions.

The following table shows the conditions under which read accesses from TMC registers return valid values. Reads at other times return **UNKNOWN** values. An x indicates that any value is permitted.

Table 4-2: Conditions for read accesses to TMC registers

Register	Name	Offset	CTL.TraceCaptEn	STS.TMCReady	MODE	ITCTRL	Remarks
RSZ	RAM Size register	0x004	x	x	x	x	-
STS.MemErr (ETR)	Status Register	0x00C	x	x	x	0	-
STS.Empty	Status Register	0x00C	1	x	x	0	-
STS.FtEmpty	Status Register	0x00C	1	x	x	0	-
STS.TMCReady	Status Register	0x00C	x	x	x	0	-
STS.Triggered	Status Register	0x00C	1	x	CB	0	-
STS.Triggered	Status Register	0x00C	0	x	x	0	Value of this bit when trace capture stops is held
STS.Full	Status Register	0x00C	x	x	x	0	Value of this bit when trace capture stops is held
RRD	RAM Read Data Register	0x010	0	1	x	0	-
RRD	RAM Read Data Register	0x010	1	x	SWF1	0	If trace memory is empty, the data that is returned is 0xFFFFFFFF.
RRD	RAM Read Data Register	0x010	1	1	CB	0	If trace memory is empty, the data that is returned is 0xFFFFFFFF.
RRP (ETB,ETF,ETR)	RAM Read Pointer Register	0x014	1	x	SWF1, SWF2	0	-
RRP (ETB,ETF,ETR)	RAM Read Pointer Register	0x014	1	1	CB	0	-
RRP (ETB,ETF,ETR)	RAM Read Pointer Register	0x014	0	1	x	0	-
RWP (ETB,ETF,ETR)	RAM Write Pointer Register	0x018	1	x	SWF1, SWF2	0	-
RWP (ETB,ETF,ETR)	RAM Write Pointer Register	0x018	1	1	CB	0	-
RWP (ETB,ETF,ETR)	RAM Write Pointer Register	0x018	0	1	x	0	-
TRG	Trigger Counter Register	0x01C	1	x	CB	0	The trigger counter is active only in Circular buffer mode
TRG	Trigger Counter Register	0x01C	0	1	x	0	The trigger counter is active only in Circular buffer mode
CTL	Control Register	0x020	x	x	x	0	-
RWD (ETB,ETF,ETR)	RAM Write Data Register	0x024	Write-only	Write-only	Write-only	Write-only	Write-only
MODE	Mode Register	0x028	1	x	x	0	-
LBUFLEVEL (ETB,ETF,ETR)	Latched Buffer Fill Level	0x02C	1	x	x	0	-
LBUFLEVEL (ETB,ETF,ETR)	Latched Buffer Fill Level	0x02C	0	1	x	0	Value of this register when trace capture stops is held
CBUFLEVEL (ETB,ETF,ETR)	Current Buffer Fill Level	0x030	1	x	x	0	-
CBUFLEVEL (ETB,ETF,ETR)	Current Buffer Fill Level	0x030	0	1	x	x	Value of this register when trace capture stops is held

Register	Name	Offset	CTL.TraceCaptEn	STS.TMCReady	MODE	ITCTRL	Remarks
BUFWM (ETB,ETF,ETR)	Buffer Level Water Mark	0x034	x	x	x	x	Programmed registers can be read at any time. The return value is the value that was programmed.
RRPHI (ETR)	RAM Read Pointer High Register	0x038	1	x	SWF1, SWF2	0	-
RRPHI (ETR)	RAM Read Pointer High Register	0x038	1	1	CB	0	-
RRPHI (ETR)	RAM Read Pointer High Register	0x038	0	1	x	0	-
RWPHI (ETR)	RAM Write Pointer High Register	0x03C	1	0	SWF1, SWF2	0	-
RWPHI (ETR)	RAM Write Pointer High Register	0x03C	1	1	CB	0	-
RWPHI (ETR)	RAM Write Pointer High Register	0x03C	0	1	x	0	-
AXICTL (ETR)	AXI Control Register	0x110	x	x	x	x	-
DBALO (ETR)	Data Buffer Address Low Register	0x118	x	x	x	x	-
DBAHI (ETR)	Data Buffer Address High Register	0x11C	x	x	x	x	-
RURP (ETR)	RAM Update Read Pointer Register	0x120	Write-only	Write-only	Write-only	Write-only	Write-only
FFSR	Formatter and Flush Status Register	0x300	x	x	x	0	-
FFCR	Formatter and Flush Control Register	0x304	x	x	x	0	-
PSCR	Periodic Synchronization Counter Register	0x308	x	x	x	0	-
AXICTL1 (ETR)	AXI Control Register 1	0xED0	x	x	x	0	-
ITATBMDATA0 (ETF)	Integration Test ATB Transmitter Data 0 Register	0xED0	Write-only	Write-only	Write-only	Write-only	Write-only
ITATBMCTR2 (ETF)	Integration Test ATB Transmitter Interface Control 2 Register	0xED4	x	x	x	1	-
ITATBMCTR1 (ETF)	Integration Test ATB Transmitter Interface Control 1 Register	0xED8	Write-only	Write-only	Write-only	Write-only	Write-only
ITATBMCTR0 (ETF)	Integration Test ATB Transmitter Interface Control 0 Register	0xEDC	Write-only	Write-only	Write-only	Write-only	Write-only
ITEVTINTR	Integration Test Event & Interrupt Status Register	0xEE0	Write-only	Write-only	Write-only	Write-only	Write-only
ITTRFLIN	Integration Test Trigger In and Flush In Register	0xEE8	x	x	x	1	-

Register	Name	Offset	CTL.TraceCaptEn	STS.TMCReady	MODE	ITCTRL	Remarks
ITATBDATA0	Integration Test ATB Data Register 0	0xEEC	x	x	x	1	-
ITATBCTR2	Integration Test ATB Control 2 Register	0xEF0	Write-only	Write-only	Write-only	Write-only	Write-only
ITATBCTR1	Integration Test ATB Control 1 Register	0xEF4	x	x	x	1	-
ITATBCTR0	Integration Test ATB Control 0 Register	0xEF8	x	x	x	1	-
ITCTRL	Integration Mode Control Register	0xF00	x	x	x	x	-
CLAIMSET	Claim Tag Set Register	0xFA0	x	x	x	x	-
CLAIMCLR	Claim Tag Clear Register	0xFA4	x	x	x	x	-
AUTHSTATUS	Authentication Status Register	0xFB8	x	x	x	x	-
DEVARCH	Device Architecture Register	0xFBC	x	x	x	x	-
DEVID1	Device Configuration Register	0xFC4	x	x	x	x	-
DEVID	Device Configuration Register	0xFC8	x	x	x	x	-
DEVTYPE	Device Type Identifier Register	0xFCC	x	x	x	x	-
PIDR4-7	Peripheral ID Registers 4-7	0xFD0-0xFDC	x	x	x	x	-
PIDR0-3	Peripheral ID Registers 0-3	0xFE0-0xFEC	x	x	x	x	-
CIDR0-3	Component ID Registers 0-3	0xFF0-0xFFC	x	x	x	x	-

4.8.2 Clock and reset

The SoC-600 TMC has a single clock input clk and an active-LOW reset input reset_n.

The reset_n signal resets all interfaces and control registers except for some of the memory mapped control registers. See the appropriate Register summary for your chosen TMC component for details of registers that are not initialized on reset and must be programmed before enabling TMC trace capture.

Note

An ETR component can be in a different power or reset domain from the AXI subordinate to which it connects. In this scenario, a CoreLink™ ADB-400 AMBA® Domain Bridge is required to cross the power or clock domain boundary.

4.8.3 Interfaces

The TMC has several interfaces.

- Debug APB interface
- ATB receiver interface
- ATB transmitter interface
- AXI manager interface
- AXI-Stream transmitter interface
- Low-power interface
- Event interfaces
- Buffer interrupt interface
- Authentication interface
- DFT interface

The following table shows the availability of each interface as a function of the configurations.

Table 4-3: Interface availability on TMC components

Interfaces	ETB	ETF	ETR	ETS
Debug APB	Present	Present	Present	Present
ATB receiver	Present	Present	Present	Present
ATB transmitter	-	Present	-	-
Memory BIST	Present	Present	-	-
AXI manager	-	-	Present	-
AXI-Stream transmitter	-	-	-	Present
Low-power	Present	Present	Present	Present
Event	Present	Present	Present	Present
Buffer Interrupt	-	-	Present	Present
Authentication	-	-	Present	-
DFT	Present	Present	Present	Present

4.8.3.1 Debug APB interface

The debug APB interface programs the registers and reads the trace data from local SRAM or system AXI.

The debug APB interface is compliant with the AMBA® APB5 protocol.

4.8.3.2 ATB receiver interface

The ATB receiver interface receives the trace data. It can support a configurable data width, ATB_DATA_WIDTH, of 32, 64 or 128 bits.

The interface can connect to a replicator, a trace source, or any other component with a standard ATB transmitter. The interface complies with the *AMBA® ATB Protocol Specification*.

4.8.3.3 ATB transmitter interface

The ATB transmitter interface is present only in ETF component and allows draining of trace data from local SRAM.

The interface can connect to a replicator, trace sinks (TPIU, ETB, or ETR), or any other component with a standard ATB receiver. The interface complies with the *AMBA® ATB Protocol Specification*.

4.8.3.4 MBIST interface

The Embedded Trace Buffer (ETB) and Embedded Trace FIFO (ETF) components include SRAM instances within the CoreSight SoC-600 TMC module. An Arm MBIST interface supports at-speed testing of the RAMs in a production test environment.

RAM cells are integrated to provide local trace buffering. The MBIST interface supports only production MBIST testing. On-line MBIST testing is not supported.

Following MBIST test, the contents of the RAMs are determined by the specific MBIST controller that you have integrated. Any previous data in the RAMs is lost unless integrating an MBIST controller that reinstates the previous memory data. MBIST controller design is beyond the scope of this document.

Partitioning the RAM into physical banks, reduces the test duration because it tests all physical RAMs in parallel. Each RAM bank has its own dedicated MBIST controller. If a single RAM bank is integrated, then you must integrate a single MBIST controller.

CoreSight SoC-600 provides the MBIST interface file and a testbench. See the *Arm® CoreSight™ System-on-Chip SoC-600 Configuration and Integration Manual*.

4.8.3.5 AXI manager interface

The ETR component has no internal memory, so it uses an AXI manager interface to access an external memory device.

The AXI manager interface can connect to an AXI interconnect for accessing system memory through a memory controller, or it can connect to any other AXI subordinate in the system. For system memory, where address translation is managed with an SMMU, the address translation can be signaled in advance by using AXI stash translation transactions.


Note

The ETR does not generate out-of-order transactions. Therefore, the AXI ID for all data write transactions is identical. However, for stash translation transactions, the AXI ID is different.

The AXI manager interface supports up to 32 outstanding write transactions but it does not support multiple outstanding reads. If an error response is returned at any time, the interface stops the operation until the debugger identifies and clears the error condition.

In ETR configuration, the memory size is programmable, rather than configurable. The width of the RSZ register determines the maximum size of the trace memory. The register is 31 bits wide, allowing a maximum value of `0x40000000`, representing 4GB.

The trace memory can be located anywhere in the system address space with the start address aligned to a 4KB boundary. Some of the lower bits of AXI address signals, araddr_m and awaddr_m, are tied LOW to ensure that all accesses are aligned to the AXI data width. The number of bits to tie LOW is calculated as $\log_2$(ATB_DATA_WIDTH/8). For example, when the AXI data bus is 64 bits wide, the lower 3 bits of the address signals are tied LOW, which ensures that only 64-bit aligned accesses can occur.

4.8.3.6 AXI-Stream transmitter interface

In the ETS component, the AXI-Stream transmitter interface replaces the AXI interface that is used in the ETR component.

The interface complies with the *AMBA® AXI-Stream Protocol Specification*. It can connect to any streaming device such as an HSSTP link layer. The connection can be either direct or through an AMBA® AXI5-Stream interconnect for sending trace off-chip.

The ETS outputs only data bytes and no position or null bytes.

4.8.3.7 Low-Power interface

The TMC has a Q-Channel Low-Power Interface (LPI) for clock gating that is present in all four TMC components.

For more information, see the *AMBA® Low Power Interface Specification*.

4.8.3.8 Event interfaces

The TMC has five event interfaces that you can connect to a Cross Trigger Interface (CTI).

The event signals are:

trigin

Receiver event input that can cause a trigger event.

flushin

Receiver event input that can cause a trace flush.

full

Transmitter event signal. When the TMC is not in integration mode, the full output signal indicates the value of the STS.Full register bit.

acqcomp

Transmitter event signal. When the TMC is not in integration mode, the acqcomp output signal indicates the value of the STS.FtEmpty register bit.

flushcomp

Transmitter event signal. When the TMC is not in integration mode, the flushcomp output signal pulses HIGH when a flush request is completed downstream.

You must connect these signals to a CoreSight™ Cross Trigger Interface (CTI).

4.8.3.9 Buffer interrupt interface

With the ETS component, the TMC generates an interrupt on the bufintr signal when the STS.Full bit is set.

With the ETR component, the TMC generates an interrupt on the bufintr signal when the STS.Full bit is set.

4.8.3.10 Authentication interface

The authentication interface provides connections for the CoreSight™ authentication signals.

With the ETR component, the DBGEN and SPIDEN signals of the authentication interface affect the behavior of the TMC as follows:

- If DBGEN is LOW, no AXI accesses are possible. Any attempted AXI access behaves as if an immediate error response had been returned, and sets the Memory Error status bit in the STS register. No transactions are issued on the AXI manager port.
- If SPIDEN is LOW, no Secure AXI accesses are possible. Any attempted AXI access while AXICTL.ProtCtrlBit1 is clear behaves as if an immediate error response had been returned. This also sets the Memory Error status bit in the STS register. No transactions are issued on the AXI manager port.

The authentication interface is absent in the ETB, ETF, and ETS components. The accesses are always non-invasive for those components, so the accesses do not have to be authenticated.

The NIDEN and SPNIDEN signals are not implemented, because the TMC does not implement any non-invasive debug functionality. For a description of the relationship to authentication interface signals, see the *Arm® CoreSight™ Architecture Specification v3.0*.

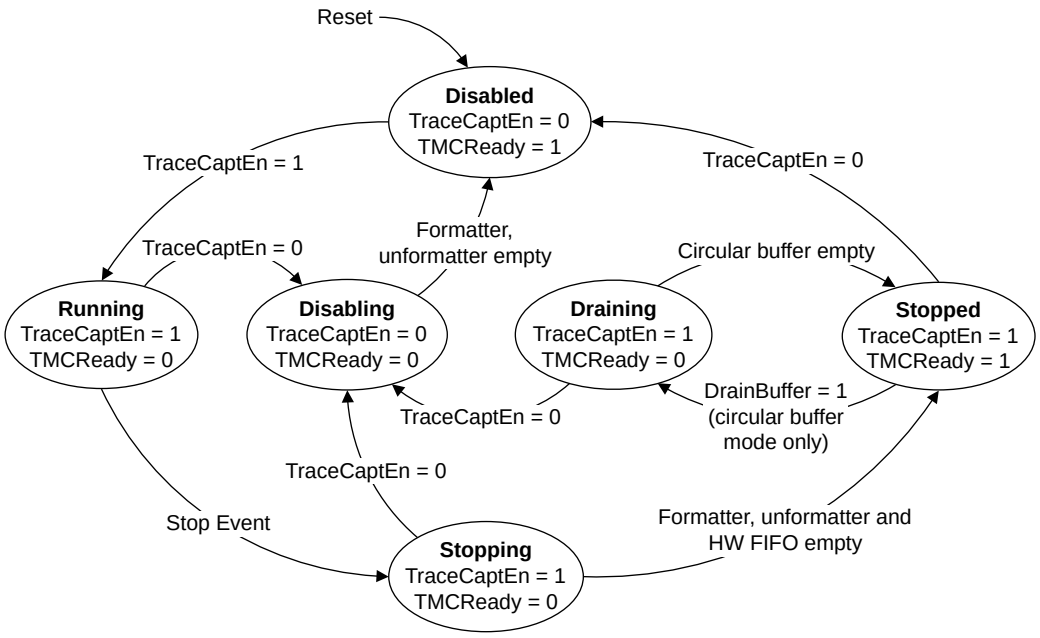
4.8.4 Operation

The TMC uses a state machine to control its operation.

4.8.4.1 Architectural state machine

The Trace Capture Enable bit, CTL.TraceCaptEn, and TMC Ready bit, STS.TMCReady, define the TMC states.

Figure 4-14: TMC architectural state machine



4.8.4.1.1 Disabled

Disabled is the default state of TMC after reset and whenever CTL.TraceCaptEn is cleared.

Disabled

CTL.TraceCaptEn=0, STS.TMCReady=1

All programming must be performed in this state.

When the TMC is in the Disabled state, the contents of most registers, including the MODE and FFCR registers, have no effect. For backwards compatibility, the contents of the circular buffer can be read in this state. The debugger must manually manage the read pointer. The TMC enters the Running state from the Disabled state when the CTL.TraceCaptEn bit is set.

4.8.4.1.2 Running

Running is the functional state where trace capture is performed.

Running

CTL.TraceCaptEn=1, STS.TMCReady=0

The Stopping state is entered from the Running state when a Stop event occurs.

4.8.4.1.3 Stopping

In the Stopping state, the TMC begins to drain the trace data from its internal pipelines to the trace memory.

Stopping

CTL.TraceCaptEn=1, STS.TMCReady=0

From the programmers model, the Stopping state is indistinguishable from the Running state.

The Stopped state is entered from the Stopping state when the following conditions are true:

- All trace has been output, including null padding, if necessary, to drain the last few bytes of trace, and the formatter and write buffer are empty.
- In Hardware FIFO (HWF) mode, the FIFO and unformatter are also empty.
- In Software FIFO mode 1 (SWF1) and Software FIFO mode 2 (SWF2), there must be space in the FIFO for data that is left in the pipeline to be written to the FIFO. To achieve this, it might be necessary to read extra data from the FIFO. If no space is available in the FIFO, then the Stopped state is not reached.
- The ETS component only supports Circular Buffer (CB) mode, so a Stopping to Stopped state transition depends on the rate at which data is accepted by the AXI5-Stream receiver device.

4.8.4.1.4 Stopped

In the Stopped state, no trace capture takes place, but data that is still in the trace memory can be read out.

Stopped

CTL.TraceCaptEn=1, STS.TMCReady=1

When in the Stopped state:

- In Circular Buffer (CB) mode in ETB, ETF, and ETR configurations, except when streaming, the captured trace can be read out over debug APB.
- In CB mode in ETF configuration, a drain can be initiated by setting the FFCR.DrainBuffer bit.
- In Software FIFO mode 1 (SWF1), the remaining contents of the FIFO can be read out over debug APB.

- In Software FIFO mode 2 (SWF2), a functional controller, such as USB, can directly read the remaining contents of the FIFO directly from the system memory.

The Disabled state is entered from this state by clearing CTL.TraceCaptEn bit.

The ETS and an ETR that is configured for streaming, do not support reading of trace data over APB.

4.8.4.1.5 Draining

The Draining state is only applicable for the ETF component.

Draining

CTL.TraceCaptEn=1, STS.TMCReady=0

In the Draining state, the contents of the buffer, that is captured in CB mode, are drained over the ATB transmitter interface. The TMC returns to the Stopped state when the buffer is empty.

4.8.4.1.6 Disabling

Disabling is an emergency stop state that can be entered at any time by clearing CTL.TraceCaptEn.

Disabling

CTL.TraceCaptEn=0, STS.TMCReady=0

The Disabling state differs from the Stopping state in the following ways:

- The TMC does not attempt to empty the contents of the FIFO in Hardware FIFO (HWF) mode. Trace that is not yet output on the ATB transmitter interface is lost.
- The next transition is to the Disabled state, not the Stopped state. This transition means that:
  - In Circular Buffer (CB) mode, while the trace can still be read over the APB for ETB, ETF, and ETR nonstreaming components, the drain operation is not possible. The drain operation is only available with the ETF component in HWF mode.
  - In Software FIFO mode 1 (SWF1) and Software FIFO mode 2 (SWF2), unretrieved trace is lost.
- Exit from the Disabling state is not dependent on reads performed from the RRD register, in SWF1 and SWF2 modes, or the ATB transmitter interface accepting writes in HWF mode. If the FIFO is full, then existing data is overwritten to enable the STOP sequence to complete. If a memory error occurs in the Disabling state, or if the STS.MemErr bit is already set and a hard stop occurs, more AXI writes are not performed. If this situation happens, the TMC discards the trace that is not output and directly moves from the Disabling to the Disabled state.

We recommend that the trace capture is stopped by programming an appropriate STOP event in the FFCR register. For example, trace capture can be stopped by setting the FFCR.StopOnFl bit, and then initiating a manual flush by setting the FFCR.FlushMan bit.

The emergency stop option is provided so that the TMC is programmer-compatible with the ETB that was delivered with SoC-400. The only way to stop trace capture when using the SoC-400 ETB was by clearing the CTL.TraceCaptEn bit. Use of the emergency stop is otherwise discouraged, especially in FIFO modes, where it can lead to loss of trace or even trace corruption.

With ETR and ETS components, the transition from the Disabling to the Disabled state is delayed indefinitely if the AXI or AXI-Stream interface is stalled indefinitely. In this case, you must clear the source of the stall, since it is not possible to abort a transfer when it has started.

4.8.4.2 Formatter and stop sequence

When EnFt in the FFCR is set, formatting is enabled.

For more information about the formatting protocol, see the *Arm® CoreSight™ Architecture Specification v3.0*.

With the ETB and ETF component, trace might be written up to 256 bits at a time. Additionally, when the formatter is enabled by setting the EnFt bit in the FFCR, a whole number of frames must be written. When stopping trace capture, the TMC pads the end of the trace so that every byte of trace that has been accepted by the TMC is written.

Note

The stop sequence might be longer than required to meet the rules, to simplify the implementation.

4.8.4.2.1 Formatter enabled

If the formatter is enabled when trace capture is stopped, then the traces are padded in the formatted frames with additional bytes of data.

The padding bytes have a value of 0x00 and an ID of 0x00. Padding continues until the following conditions are met:

- A whole number of frames have been generated.
- The trace is aligned to the memory width. This means that if the ATB interface is configured to 128 bits, then a multiple of two frames has been generated to meet the 256-bit memory width.

In HWF mode, the ATB transmitter interface does not generate any additional traces at the end of trace. However, all trace stored internally is flushed out of the TMC when trace capture is stopped.

4.8.4.2.2 Formatter disabled

Disabling the SoC-600 formatter is deprecated, and is supported in CB mode only.

If the formatter is disabled when trace capture is stopped, then the trace is padded with additional bytes so that the precise end of the trace can be determined, as follows:

- A single byte of value `0x01`, to indicate the position of the last byte before the stop sequence.
- Zero or more bytes of value `0x00`, to align to the memory width.

4.8.4.3 Trigger, flush, and stop events

The SoC-600 TMC has a Formatter and Flush Control Register (FFCR).

The FFCR includes controls for the following:

- Enabling the formatter to wrap data from multiple trace sources into frames as the *Arm® CoreSight™ Architecture Specification v3.0* describes.
- The insertion of trigger markers into the formatted trace stream.
- When to stop trace capture.
- When to perform a flush.
- Draining the trace buffer when in CB mode and in the Stopped state.

In Hardware FIFO mode (HWF), Software FIFO mode 1 (SWF1) and Software FIFO mode 2 (SWF2), setting the following bits of the FFCR results in **UNPREDICTABLE** behavior:

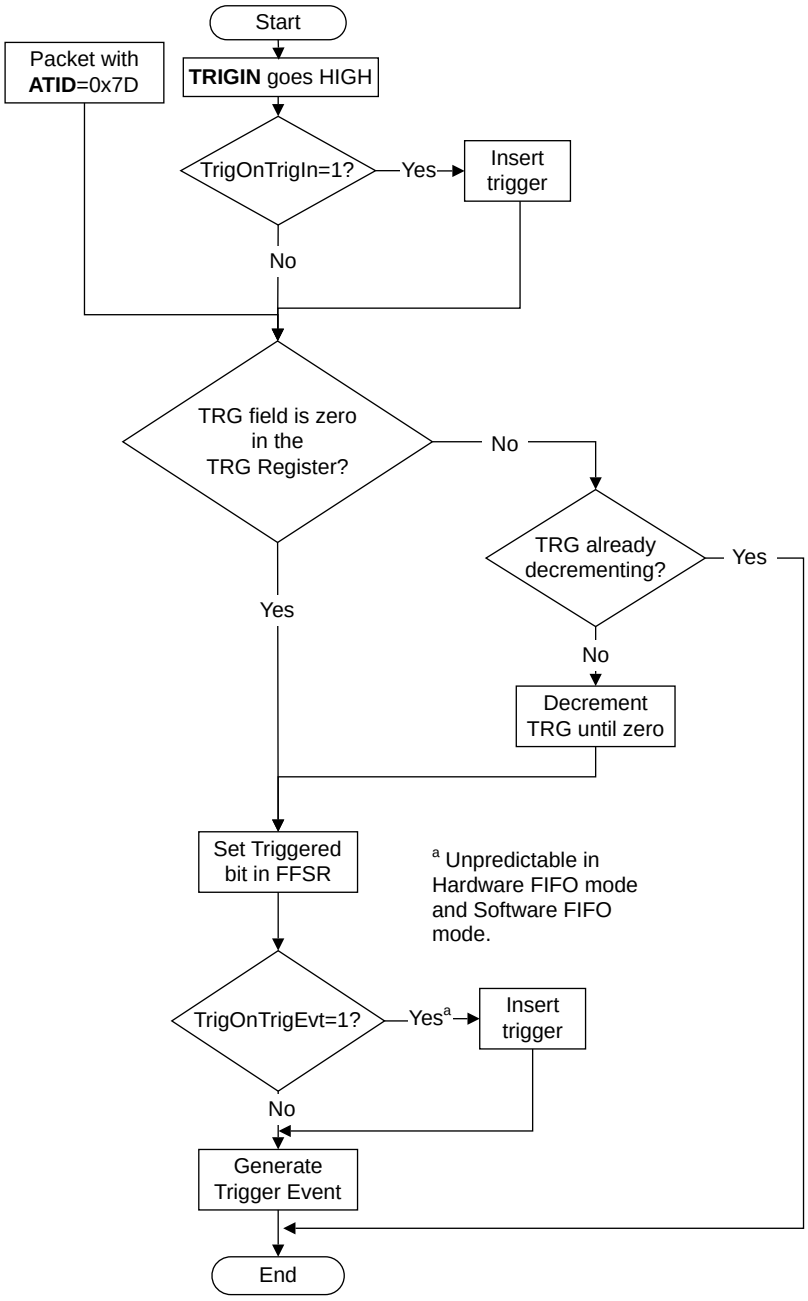
- TrigOnTrigEvt
- StopOnTrigEvt
- FOnTrigEvt

4.8.4.3.1 trigin input and ATB receiver interface trigger

The SoC-600 TMC takes action when an event is sampled on the trigin input signal, or a packet is accepted on the ATB receiver interface when ATID is equal to `0x7D`.

The following figure shows the actions taken.

Figure 4-15: Actions taken on input trigger



Note

The TMC always outputs any outstanding triggers in the trace before completing a flush or stopping trace capture. A rapid stream of triggers on the trigin input signal or on the ATB receiver interface can cause the flush or stop to be delayed.

4.8.4.3.2 Flush and stop

A trigger event is one of several events that can cause various kinds of flush, stop the TMC, and cause additional trigger insertion.

The following table shows how the TMC responds to these events.

Table 4-4: Event generation

Event	Mode	Outcome
With the ETF component, a flush request on the ATB transmitter interface	CBF or SW	Acknowledged immediately, and otherwise is ignored.
With the ETF component, a flush request on the ATB transmitter interface	HWF	Flush the trace sources that feed the TMC, and the contents of the TMC.
FlushMan or (FOnFlIn & flushin signal)	CB	Flush the trace sources that feed the TMC: • Insert a trigger if TrigOnFl is set in the FFCR. • Stop trace capture if StopOnFl is set in the FFCR.
FlushMan or (FOnFlIn & flushin signal)	HWF	Flush the trace sources that feed the TMC: • Insert a trigger if TrigOnFl is set in the FFCR. • Stop trace capture if StopOnFl is set in the FFCR.
FlushMan or (FOnFlIn & flushin signal)	SWF1 or SWF2	Flush the trace sources that feed the TMC, and ensure that the flushed trace is ready to be read by subsequent reads of the RRD register: • Insert a trigger if TrigOnFl is set in the FFCR. • Stop trace capture if StopOnFl is set in the FFCR.
Trigger event	CB	If StopOnTrigEvt is set in the FFCR, then stop trace capture. If StopOnTrigEvt is not set in the FFCR: • Insert a trigger if TrigOnFl is set in the FFCR. • Stop trace capture if StopOnFl is set in the FFCR.
Trigger event	HWF, SWF1, or SWF2	Ignored
TraceCaptEn bit cleared during trace capture	Any	Stop trace capture immediately. The captured trace is lost.

4.8.4.3.3 Common usage

Many bits can be set simultaneously, leading to a wide range of programming settings, not all of which are useful.

In practice, the most common setting in CB mode is to set the TrigOnTrigIn, FOnTrigEvt, StopOnFl, EnTI and EnFt bits in the FFCR.

1. Wait for a trigger.
2. Insert a trigger into the trace stream.
3. Count down the trigger counter.
4. Flush.

5. Stop trace capture.

4.8.5 Standard usage models

The recommended usage model for the SoC-600 TMC depends on its operating mode, which determines how the TMC captures trace data.

For a summary of each operating mode, see Trace Memory Controller.

See the following sections for the standard usage model for each operating mode:

- Circular Buffer mode
- Software FIFO mode 1
- Software FIFO mode 2
- Hardware FIFO mode

4.8.5.1 Circular Buffer mode

In Circular Buffer (CB) mode, TMC captures trace using its storage as a circular buffer, overwriting old trace when the buffer is full. No trace is output until capture is complete. In this mode, trace capture can automatically stop after receiving a trigger signal.


Note

CB mode is available for ETB, ETF, ETR, and ETS components.

Arm recommends that you follow this standard usage model when the SoC-600 TMC is capturing trace data in CB mode. CB mode has an optional manual stop.

1. Wait until STS.TMCReady equals one.
2. When using an ETR, ETF, or ETB component, program the MODE Register for CB mode.
3. Program the FFCR Register. We recommend that you set the TrigOnTrigIn, FOnTrigEvt, StopOnFl, EnTI, and EnFt bits. Setting the recommended bits enables:
   - Formatting
   - Inserting a trigger when a trigger is observed on the trigin input signal
   - Flushing and then stopping the TMC, following a delay corresponding to the value of the TRG Register
4. Program the TRG Register, to control the amount of buffer to be dedicated to the period after a trigger is observed.
5. When using an ETR, ETF, or ETB component, write RWP and write RRP with the same value as RWP. We recommend RWP=DBA for the ETR component. Otherwise, RWP=0.

6. When ETR component is used, write DBAHI, DBALO, RSZ, AXICTL, and RWPHI. Also, write RRPHI with the same value as RWPHI.
7. Set the TraceCaptEn bit in the CTL Register. Setting this bit starts the trace session.
8. Either:
   - Wait until TMCReady in STS register equals one, signaling that the trace session is over, because a trigger event has occurred.
   - To manually stop capture without a trigger event, set the FlushMan bit in the FFCR Register. Then wait until TMCReady equals one, signaling that the trace session is over, because the manual flush completed.
9. When using an ETR, ETF, or ETB component, read the contents of the trace buffer by performing successive reads to the RRD Register. Continue the reads until the value `0xFFFFFFFF` is returned.
10. Clear the TraceCaptEn bit in the CTL Register.

4.8.5.2 Software FIFO mode 1

In Software FIFO mode 1 (SWF1), the component functions as a FIFO where data is read out by software over the debug APB interface. This mode provides a low-speed communication channel for trace data reusing the existing programming interface.


Note

SWF1 mode is available for ETB, ETF, and ETR components.

Arm recommends that you follow this standard usage model when the SoC-600 TMC is capturing trace data in SWF1 mode.

1. Wait until TMCReady is equal to one.
2. Program the MODE Register for Software FIFO mode.
3. Program the FFCR Register. We recommend that you set the TrigOnTrigIn, StopOnFl, EnTI, and EnFt bits. This enables formatting, inserting a trigger when a trigger is observed on TRIGIN. To stop the TMC, perform a manual flush using FFCR.FlushMan. This flushes any remaining data out of the TMC and then stops on the flush completion due to FFCR.StopOnFl.
4. Program the BUFWM Register to the required threshold fill level. You can usually set this register to zero.
5. Program the TRG Register to control the amount of buffer to be dedicated to the period after a trigger is observed.
6. Write RWP and write RRP with the same value as RWP. We recommend RWP=DBA for the ETR component. Otherwise, RWP=0.
7. When ETR component is used, write DBAHI, DBALO, RSZ, AXICTL, and RWPHI. Also, write RRPHI with the same value as RWPHI.
8. Set the TraceCaptEn bit in the CTL Register. This starts the trace session.

9. Start reading data from the RRD Register. If the value `0xFFFFFFFF` is returned, then no data is available, and the read must be retried. Continue until the trace session is over, for example, following receipt of a trigger in the trace stream.
10. Either:
    - Read data from the FIFO to get flushed data, retrying when `0xFFFFFFFF` is returned, until TMCReady is equal to one. This indicates that the trace session is over, because a trigger event has occurred.
    - To stop capture manually without a trigger event, set the FlushMan bit in the FFCR Register to flush, then wait until TMCReady is equal to one. This indicates that the trace session is over because the manual flush completed.

```
Repeat {
  Read Data from RRD register
  If (Data = 0xFFFFFFFF) {
    Read TMCReady from STS register
    If (TMCReady = 1) {
  Stop
    }
  } else {
    Add Data to the end of the trace
  }
}
```

11. Read the remaining data from the FIFO, stopping when `0xFFFFFFFF` is returned. This indicates that the FIFO is empty.
12. Clear the TraceCaptEn bit in the CTL Register.

4.8.5.3 Software FIFO mode 2

In Software FIFO 2 mode (SWF2), the trace memory accessible through the AXI interface is used as a FIFO. The key difference from Software FIFO mode 1 is that the data is read out by the debugger directly from the memory, bypassing the ETR datapath.

The ETR still manages the memory read pointer, so the debugger must inform the ETR of the amount of trace extracted. The trace that is extracted from the memory can be output over a streaming interface, such as USB, which provides much higher bandwidth than the debug APB interface.


Note

SWF2 mode is available for ETR components only.

Arm recommends that you follow this standard usage model when the SoC-600 TMC is capturing trace data in SWF2 mode.

1. Wait until TMCReady is equal to 1.
2. Program the MODE register for SWF2 mode.

3. Program the FFCR register. We recommend that you set the TrigOnTrigIn, StopOnFl, EnTI, and EnFt bits. This enables formatting, inserting a trigger when a trigger is observed on TRIGIN. To stop the TMC, perform a manual flush using FFCR.FlushMan. This flushes any remaining data out of the TMC and then stops on the flush completion due to FFCR.StopOnFl.
4. Program the BUFWM register to the required threshold fill level. You can usually set this register to zero.
5. Program the TRG register to control the amount of buffer to be dedicated to the period after a trigger is observed.
6. Write RWP and write RRP with the same value as RWP. We recommend RWP = DBA for the ETR component. Otherwise, RWP = 0.
7. Write DBAHI, DBALO, RSZ, AXICTL, and RWPHI. Also, write RRPHI with the same value as RWPHI.
8. Set the TraceCaptEn bit in the CTL register. This starts the trace session.
9. Calculate how many Bytes are available for reading, see the two methods below, and read data directly from memory.

```
Method 1:
rsz = Read value of RSZ register
rwp = Read value of RWP registers
rrp = Read value of RRP registers
if (rwp > rrp) { numBytes = rwp - rrp }
else if (rwp < rrp) { numBytes = rsz - rrp + rwp }
else if (rwp == rrp) & (STS.Full == 1) { numBytes = rsz }
else if (rwp == rrp) & (STS.Full == 0) { numBytes = 0 }
Method 2:
numBytes = 4 × read value of CBUFLEVEL register.
```

1. After having read data directly from memory increment RRP by writing directly to the RURP register
2. Check to see if STS.TMCReady has been set to 1 automatically by the ETR, or if there is a need to stop capture. If neither of these are true, repeat from step 9.
3. If manual trace capture stop is needed, set the FFCR.FlushMan to 1, and FFCR.StopOnFl to 1
4. Repeat steps 9-11 until STS.TMCReady bit is 1 and there are no more bytes available for reading.
5. Set CTL.TraceCaptEn to 0.

4.8.5.4 Hardware FIFO mode

In Hardware FIFO mode (HWF), TMC uses its storage as a FIFO, acting as a link between a trace source and a trace sink. No trace is lost or overwritten, and backpressure is applied through the AMBA® Trace Bus (ATB) to the trace source when the FIFO becomes full.

Most trace sources eventually overflow when subject to backpressure for a long time, but it is always the trace sources that lose trace, not the ETF.

HWF mode enables large bursts of trace to be smoothed, reducing the requirement to exert backpressure, so that:

- Trace can be output over a trace port using fewer pins than would be required without the ETF, by smoothing peaks in trace bandwidth over long periods.
- A subsequent ETR receiving trace through ETF can cope with large delays that are introduced by higher priority managers on the AXI interconnect without losing the trace.


Note

HWF mode is available for ETF components only.

Arm recommends that you follow this standard usage model when the SoC-600 TMC is capturing trace data in Hardware FIFO (HWF) mode.

1. Wait until TMCReady is equal to one.
2. Program the MODE register for HWF mode.
3. Program the FFCR register. We recommend that you set the EnTI and EnFt bits. This enables formatting, inserting a trigger when a trigger is observed on TRIGIN. It is not usually necessary to set any other bits in this mode because the trace sink performs flush and stop control.
4. Program the BUFWM register to the required threshold fill level. You can usually set this register to zero.
5. Write RWP and write RRP with the same value as RWP. We recommend RWP=`0x0`.
6. Set the TraceCaptEn bit in the CTL register. The TMC is now in HWF mode.

4.9 Trace Port Interface Unit

An external Trace Port Analyzer (TPA) captures trace data when the TPIU drives the external pins of a trace port.

The TPIU does the following:

- Coordinates stopping trace capture when a trigger is received.
- Inserts source ID information into the trace stream so that trace data can be re-associated with its trace source. The operation of the trace formatter is described in the *Arm® CoreSight™ Architecture Specification v3.0*.
- Outputs the trace data over trace port pins.
- Outputs patterns over the trace port so that a TPA can tune its capture logic to the trace port, maximizing the speed at which trace can be captured.

4.9.1 Clocks and resets

The clock and reset signals of the TPIU are clk, traceclkin, reset_n, and treset_n.

The TPIU includes an asynchronous bridge between the traceclkin clock domain and the rest of the design.

An external APB asynchronous bridge can be used to bridge to another clock domain if necessary.

4.9.2 Functional interfaces

The SoC-600 TPIU has functional interfaces.

The functional interfaces are:

ATB receiver interface
Receives trace data

APB completer interface
Accesses the TPIU registers

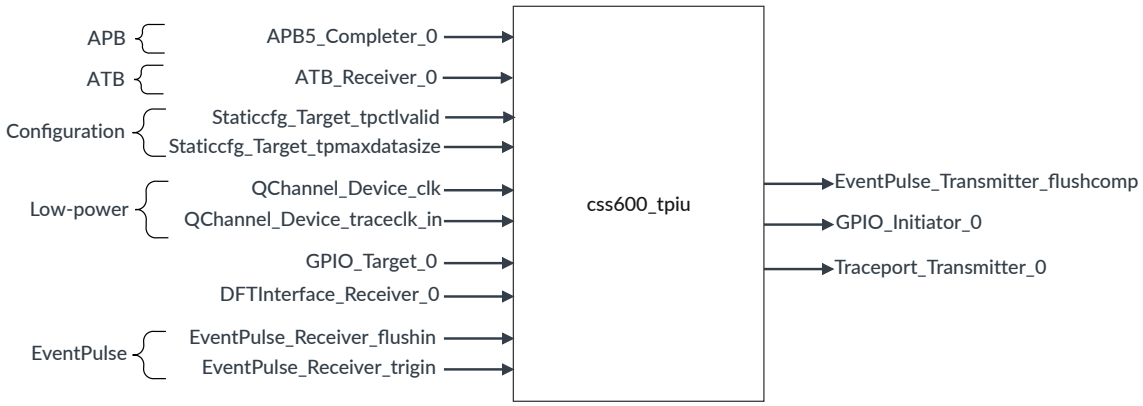
Trace out port
Connects to the external trace port pins

trigin and flushin event interfaces
Implement synchronizers so that they can be connected to a CTI in a different clock domain

The TPIU also supports the extctlin[7:0] and extctlout[7:0] signals. These signals are designed to enable debug tools to multiplex the pins used by the trace out port with other functions.

The following figure shows the external connections of the TPIU.

Figure 4-16: Trace Port Interface Unit block diagram



4.9.3 Parallel trace out port

The SoC-600 TPIU supports parallel trace output.

The SSPSR register indicates the supported parallel trace port widths. To select the active parallel trace port width, program the CSPSR register.

Trace out port signals

The following table shows the trace out port signals.

Table 4-5: Trace out port signals

Signal	Type	Description
traceclk	Output	Output clock, that the Trace Port Analyzer (TPA) uses to sample the other pins of the trace out port. This signal runs at half the speed of traceclkin, and data is valid on both edges of this clock.
tracedata[31:0]	Output	Output data. A system might not connect all the bits of this signal to the trace port pins. The connection depends on the number of pins available and the bandwidth that is required to output trace.
tracectl	Output	Signal to support legacy TPAs which cannot support formatter operation in continuous mode. Connection of this signal to a pin is optional.

For more information about these signals, see the *Arm® CoreSight™ System-on-Chip SoC-600 Configuration and Integration Manual*.

4.9.4 traceclk alignment

The SoC-600 TPIU does not offset the edges of the traceclk signal from the edges of the trace data signals tracedata and tracectl. For compatibility with the maximum number of TPAs, we recommend that you delay traceclk so its edges are in the middle of the stable phases of the data signals.

To support the widest range of targets at the maximum speed, we recommend that TPAs support systems with a variety of alignments of traceclk relative to the data signals. This recommendation is also applicable to systems where the edges of traceclk occur at the same time as transitions of the data signals.

4.9.5 tracectl removal

The TPIU supports traceclk + tracedata + tracectl, with a minimum tracedata width of 2, and traceclk + tracedata, with a minimum data width of 1.

The chosen mode depends on the connected Trace Port Analyzer (TPA) or capture device. Legacy capture devices use the control pin to indicate when there is valid data to capture. Newer capture devices can use more pins for data and do not require a reserved tracectl pin.

If support for legacy TPAs is not required, it is not necessary to implement the tracectl signal. The FFSR.TCPresent bit indicates whether a TPIU includes the tracectl signal.

4.9.6 tracectl encoding

When tracectl is implemented, and bypass or normal mode is selected in the FFCR register, the encodings of tracectl and tracedata[1:0] are designed to be backwards compatible with systems designed without CoreSight™, where a trace port is driven by a single ETM.

The encodings indicate to the Trace Port Analyzer (TPA):

- Whether the trigger has occurred. This can be used by the TPA to stop trace capture, when the TPA is responsible for stopping trace capture.
- Whether to capture data from the trace port in this cycle.

4.9.7 Trace port triggers

The SoC-600 TPIU trace port is backwards compatible with non-CoreSight™ systems where a single ETM drives the trace port. Compatibility is achieved when the tracectl signal is implemented, and bypass or normal mode is selected in the FFCR register.

The trigger is an indication to the TPA to stop trace capture. In CoreSight systems, the TPIU receives trigger events from trace sources through the cross-triggering system. The TPIU sends a trigger event over the trace out port to the TPA when it is ready for trace capture to stop.

The TPIU might signal a trigger as a result. This trigger can be:

- Directly from an event such as a pin toggle from the CTI.
- A delayed event such as a pin toggle that has been delayed by the Trigger Counter register values. See TCVR and TCMR.
- The completion of a flush.

The following table extends the *Arm® Embedded Trace Macrocell Architecture Specification ETMv1.0 to ETMv3.5* on how a trigger is represented.

Table 4-6: CoreSight representation of triggers

tracectl	tracedata[1]	tracedata[0]	Trigger Yes/ No	Capture Yes/ No	Description
0	x	x	No	Yes	Normal trace data
1	0	0	Yes	Yes	Trigger packet. The trigger packet encoding is not generated by the TPIU.
1	1	0	Yes	No	Trigger
1	x	1	No	No	Trace disable

4.9.7.1 Correlation with afvalid

When the TPIU receives a trigger signal, it can request a flush of all trace components through the ATB receiver interface.

This flush request depends on the settings in the FFCR register. The flush request causes all information around the trigger event to be flushed from the system before normal trace information is resumed. The flush ensures that all information that is related to the trigger is output before the TPA, or other capture device, is stopped.

With FFCR.FOnTrig set to 1, it is possible to indicate the trigger on completion of the flush routine. Knowing the trigger ensures that if the TPA stops the capture on a trigger, the TPA gets all historical data relating to the trigger.

4.9.8 Programming the TPIU for trace capture

There are several points that you should consider when programming the SoC-600 TPIU registers for trace capture.

The points for consideration are:

- TPAs that are only capable of operation with tracectl must only use the formatter in either bypass or normal mode, not in continuous mode.
- Following a trigger event within a multi-trace source configuration, we recommend performing a flush to ensure output of all historical information that is related to the trigger.
- If Flush on Trigger Event and Stop on Trigger Event options are chosen, then the TPA does not capture any data after the trigger. When the TPIU is instructed to stop, it discards any subsequent trace data, including data returned by the flush. Select Stop on Flush completion instead.
- Multiple flushes can be scheduled using Flush on Trigger Event, Flush on flushin, and manual flush. When one of these requests is made, it masks more requests of the same type. This masking means that repeated writing to the manual flush bit does not schedule multiple manual requests unless each is permitted to complete first.
- Unless multiple triggers are required, it is not advisable to set both Trigger on Trigger Event and Trigger on Flush Completion, if Flush on Trigger Event is also enabled. In addition, if Trigger on trigin is enabled with this configuration, it can also cause multiple trigger markers from one trigger request.
- We recommend that you only enable the pattern generator while the formatter is stopped.

4.9.9 Example configuration scenarios

The example configuration scenarios show different ways of using the SoC-600 TPIU.

The example scenarios are:

- Capturing trace after an event, and stopping

- Only indicate triggers and continue to flush
- Multiple trigger indications
- Independent triggering and flushing

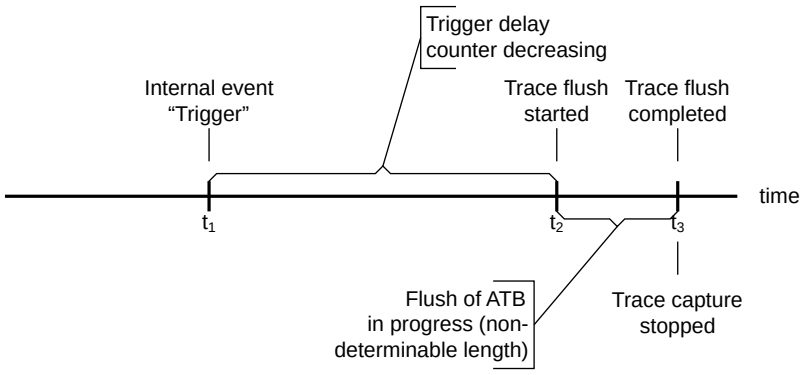
4.9.9.1 Capturing trace after an event, and stopping

A minimum amount of time must elapse before a trace capture can be stopped.

The elapsed time between the trigger and the stopping of the trace must be long enough to allow the trace data to progress through the system. Any historical information, relating to previous events, must have been emitted.

The following figure shows a possible time-line of events where an event of interest, referred to as a trigger event, causes some trace to be captured. When the trace has been captured, the trace capture device can be stopped.

Figure 4-17: Capturing trace after an event and stopping



When one trace source is used, you do not have to flush the system. Instead, you can increase the length of the trigger counter delay to enable more trace to be generated, effectively pushing out historical information.

The trigger event at time t1 is signaled to the TPIU through the cross-triggering system. The trace source that generated the trigger event might also embed some trigger information in its trace stream at this point. The TPA only registers a trigger at time t3, when it is safe to stop trace capture. If the TPIU is in bypass or normal mode, it embeds a trigger in the formatted trace stream at time t3, and signals a trigger on tracectl.

In the figure, the action that causes trace capture to be stopped at time t3 can be one of the following:

- The TPA can watch for a trigger to be indicated through tracectl and stop.
- The TPA can watch for a trigger to be indicated in the tracedata stream, using continuous mode without the requirement for tracectl.
- The TPIU can automatically stop trace after it has signaled the trigger to the TPA.

4.9.9.2 Only indicate triggers and continue to flush

You can indicate a trigger at the soonest possible moment, and cause a flush, while at the same time permitting externally requested flushes.

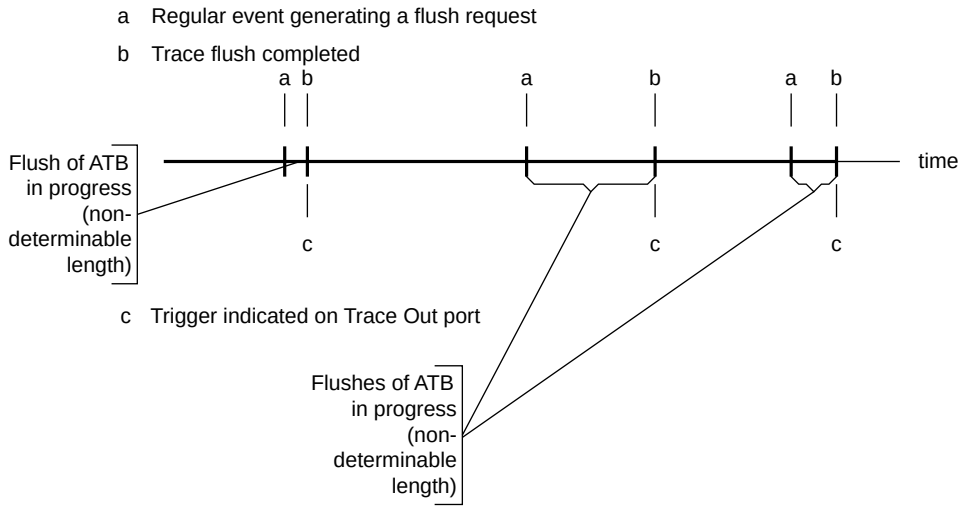
This ability enables trace around a key event to be captured, and all historical information to be stored within a period immediately following the trigger. Use a secondary event to cause regular trace flushes.

4.9.9.3 Multiple trigger indications

Sending a trigger to external tools can have extra consequences apart from stopping trace capture.

For example, this can be in cases where the events immediately before the trigger might be important, but only a small buffer is available. In this case, uploads to a host computer for decompression can occur, reducing the amount of trace data that is stored in the TPA. This procedure is also useful where the trigger originated from a device that is not directly associated with a trace source, and is a marker for a repeating interesting event.

Figure 4-18: Multiple trigger indications from flushes



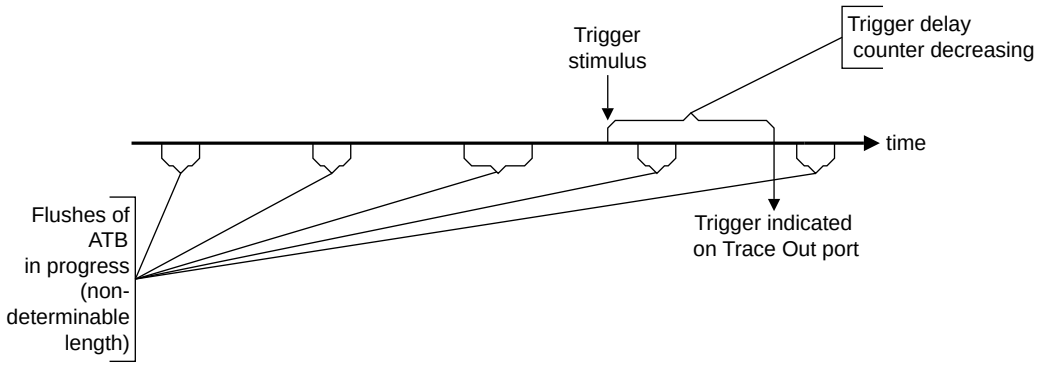
4.9.9.4 Independent triggering and flushing

The SoC-600 TPIU has separate inputs for flushes and triggers.

Although flushes can generate triggers, and triggers generate flushes, there might be a requirement to keep them separate. A timing block that is connected to a CTI can provide a consistent flow of new information through the Trace Out port by scheduling a regular flush. These regular events must not be marked in the trace stream as triggers.

Special events coming through the CTI that require a marker must be passed through the trigin pin. These events can either be indicated immediately or, as the following figure shows, they can be delayed through other flushes and then indicated to the TPA.

Figure 4-19: Independent triggering during repeated flushes



4.9.10 TPIU pattern generator

A simple set of defined bit sequences or patterns can be output over the trace port. The TPA, or other associated trace capture device, can detect these sequences.

Analysis of the output can indicate whether it was possible to increase or, for reliability, to decrease the trace port clock speed. To ensure reliable data capture, the patterns can be used to determine the timing characteristics. The patterns can also be used for measuring the timing characteristics on the data channels to the TPA to determine whether data can be captured reliably.

Note

We recommend that you only enable the pattern generator while the formatter is stopped.

4.9.10.1 Pattern generator modes of operation

To enable various metrics to be determined, several patterns are supported.

The metrics can include:

- Timing between pins
- Data edge timing
- Voltage fluctuations
- Ground bounce
- Crosstalk

When examining the trace port, you can choose from the following pattern modes:

Timed

Each pattern runs for a programmable number of traceclkin cycles. After completing the programmed number of cycles, the pattern generator unit reverts to an off state where normal trace is output, assuming trace output is enabled. The first thing that the trace port outputs after returning to normal trace is a synchronization packet. This behavior is useful with special trace port analyzers and capture devices that are aware of the pattern generator. The TPIU can be set to a standard configuration that the capture device expects. The preset test pattern can then be run, after which the TPA is calibrated ready for normal operation. The TPIU switches to normal operation automatically, without the requirement to reprogram the TPIU.

Continuous

The selected pattern runs continuously until manually disabled. This behavior is primarily intended for manual refinement of electrical characteristics and timing.

Off

When neither of the other two modes is selected, the device reverts to outputting any trace data. After timed operation finishes, the pattern generator reverts to the off mode.

4.9.10.2 Supported options

Patterns operate over all the tracedata pins for a given port width setting.

Test patterns are aware of port sizes and always align to tracedata[0]. Walking bit patterns wrap at the highest data pin for the selected port width even if the device has a larger port width available. Also, the alternating patterns do not affect disabled data pins on smaller trace port sizes.

4.9.10.3 Walking 1s

All output pins are clear, with only a single bit set at a time, tracking across every tracedata output pin.

This pattern can be used to watch for data edge timing, synchronization, high-voltage level of logic 1, and cross talk against adjacent wires. Walking 1s can also be used as a simple way to test for broken or faulty cables and data signals.

4.9.10.4 Walking 0s

All output pins are set, with only a single bit cleared at a time, tracking across every tracedata output pin.

Like the walking 1s, walking 0s can be used to watch for data edge timing, synchronization, low voltage level of logic 0, crosstalk, and ground lift.

4.9.10.5 Alternating 55/AA pattern

Alternate tracedata pins are set with the others clear.

This alternates every cycle with the sequence starting with tracedata[0] set to 55 pattern = `0b0101_0101`, followed by tracedata[1] set to AA pattern = `0b1010_1010`.

The pattern repeats over the entire selected bus width. This pattern can be used to check voltage levels, crosstalk, and data edge timing.

4.9.10.6 Alternating FF/00 pattern

On each clock cycle, the tracedata pins are either all set FF pattern or all cleared 0x00 pattern.

This sequence of alternating the entire set of data pins is a good way to check the power supply stability to the TPIU and the final pads, because of the stresses the drivers are under.

4.10 CoreSight Address Translation Unit

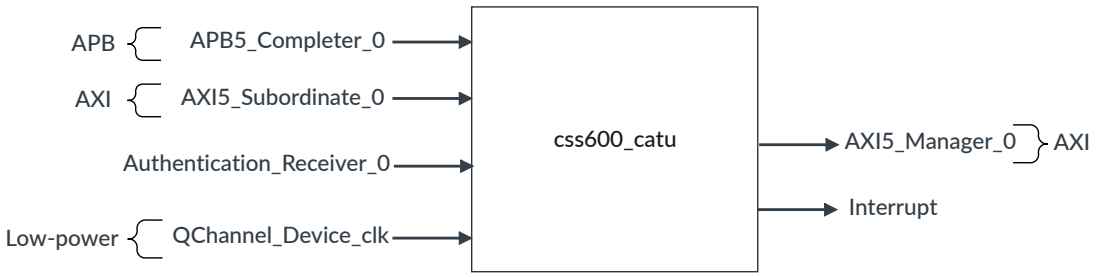
The purpose of the `css600_catu` CoreSight™ Address Translation Unit (CATU) is to translate addresses between an AXI manager and system memory.

The SoC-600 CATU is normally used along with the Trace Memory Controller (TMC) to implement scattering of virtual trace buffers in physical memory.

The CATU translates contiguous Virtual Addresses (VAs) from an AXI manager into non-contiguous Physical Addresses (PAs) that are intended for system memory.

The following figure shows the external connections on the CATU.

Figure 4-20: css600_catu logical connections



4.10.1 CATU low-power interface

The SoC-600 CoreSight™ Address Translation Unit (CATU) has a Q-Channel low-power interface for clock gating.

For more information, see the *AMBA® Low Power Interface Specification*.

4.10.2 CATU APB completer interface

The CoreSight™ Address Translation Unit (CATU) APB interface is used for programming the registers.

The interface is compliant with the AMBA® APB5 protocol. The interface only supports 32-bit-wide accesses.

For more information, see the *AMBA® APB Protocol Specification*.

4.10.3 CATU AXI manager

The AXI manager interface can be connected to an AXI interconnect for accessing system memory through a memory controller. The interface can also be connected directly to any other AXI subordinate in the system.

The AXI manager interface complies with the AMBA® 5 AXI protocol. For more information, see the *AMBA® AXI Protocol Specification*.

In Pass-through mode, all AXI transactions that are received on the AXI subordinate interface go to the AXI manager interface without any modification.

In Translate mode, the interface issues internally generated read transactions to fetch translation data. For more information, see Scatter list.

The AXI subordinate interface supports up to 16 outstanding write transactions and 16 outstanding read transactions. The AXI manager interface supports up to 16 outstanding write transactions and 17 outstanding read transactions.

4.10.4 CATU AXI subordinate

The AXI subordinate interface can be connected to an AXI interconnect. The interface can also be connected directly to any other AXI manager in the system.

The AXI subordinate interface complies with the AMBA® 5 AXI protocol. For more information, see the *AMBA® AXI Protocol Specification*.

The following optional features of the AXI protocol specification are not supported:

AXI out of order transaction model

Ports AWID, ARID, WID, RID, and BID are not implemented. All write transactions are expected to complete in order.

AXI Quality of Service (QoS) signaling

Ports AWQOS and ARQOS are not implemented.

Multiple-region signaling

Ports AWREGION and ARREGION are not implemented.

User-defined signaling

Ports AWUSER, ARUSER, WUSER, RUSER, and BUSER are not implemented.

4.10.5 CATU interrupt interface

The CoreSight™ Address Translation Unit (CATU) generates an interrupt when the incoming AXI address transaction is not in the valid address range and the interrupt is enabled.

For more information, see the IRQEN register.

The interrupt signal remains asserted while the CATU is enabled. Disabling the CATU, by setting CONTROL.ENABLE to 0, is the only way to clear the interrupt signal.

4.10.6 CATU Authentication interface

The Authentication interface provides connections for the CoreSight™ Authentication signals.

There are two signals, dbgen and spiden, that are both required to enable Secure AXI debug accesses. If dbgen is used on its own, that is dbgen = 1 and spiden = 0, then only Non-secure AXI debug accesses are allowed.

The dbgen and spiden authentication signals must not be changed unless the CATU receiver interface has no outstanding transactions, that is when STATUS.READY == 1.

For more information, see the *Arm® CoreSight™ Architecture Specification v3.0*.

4.10.7 Software interfaces

Software can interact with the CATU through:

- A set of control and status registers. For more information, see CoreSight Address Translation Unit.
- A scatter list of memory-based data structures that define physical addresses.

4.10.7.1 Scatter list

The CATU uses a scatter list in memory to find the addresses of pages of memory to use for trace storage.

All the physical memory pages that are made available for trace data storage are 4KB. Each entry in the scatter list points to one of these 4KB pages.

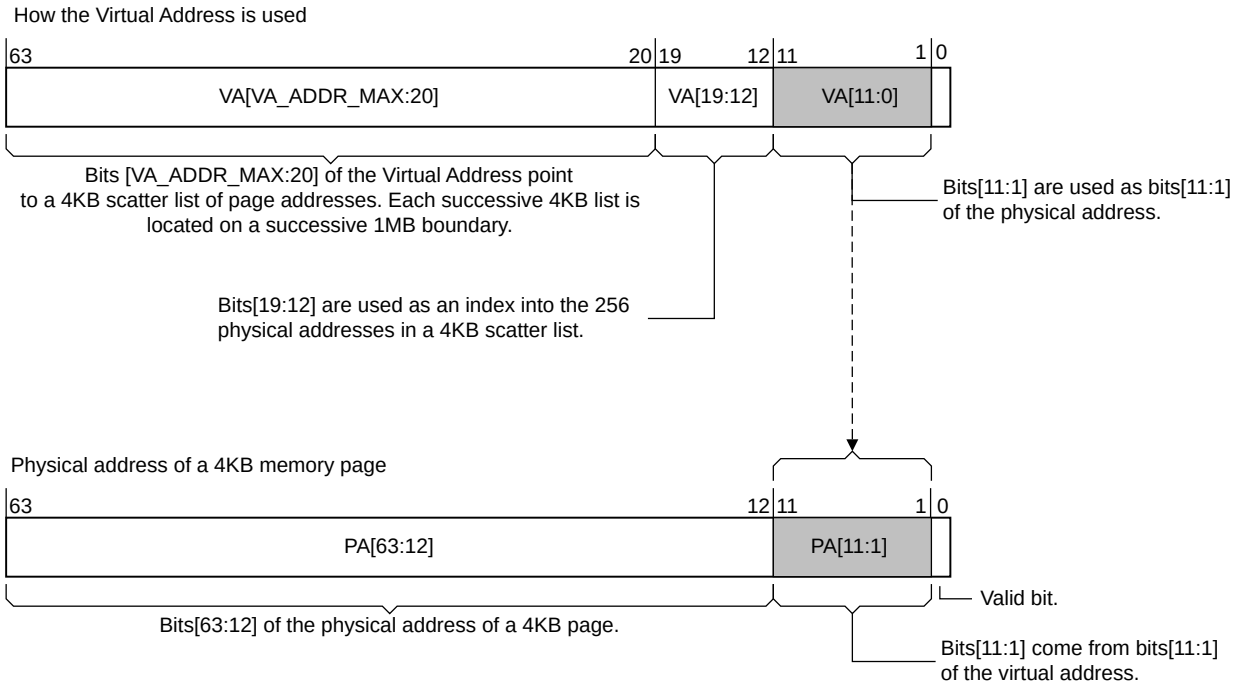
The scatter list comprises a series of 4KB address regions, each region split into 2 × 2KB subpages. The base address of the scatter list is defined in the SLADDRHI and SLADDRLO registers. At reset, the INADDRHI and INADDRLO registers contain a VA reference to the physical address that is defined in the SLADDRHI and SLADDRLO registers.

Each bottom 2KB subpage holds up to 256 64-bit physical addresses, each one pointing to a 4KB trace storage page in physical memory. Each top 2KB subpage holds only two addresses, the address of the next linked list and the address of the previous linked list. The address of the next linked list is the last entry in a 4KB list of pages, and the address of the previous linked list is the last-but-one entry. The total VA space that is covered by any 4KB scatter list is 256 entries × 4KB pages = 1MB.

Scatter list address format

Each entry in the bottom 2KB subpage link list is made up as follows:

Figure 4-21: Structure of addresses that are used in VA → PA translation



The addresses are used as follows:

1. The top [VA_ADDR_MAX:20] bits of the virtual address are used to locate the required 4KB list of 4KB page physical addresses.

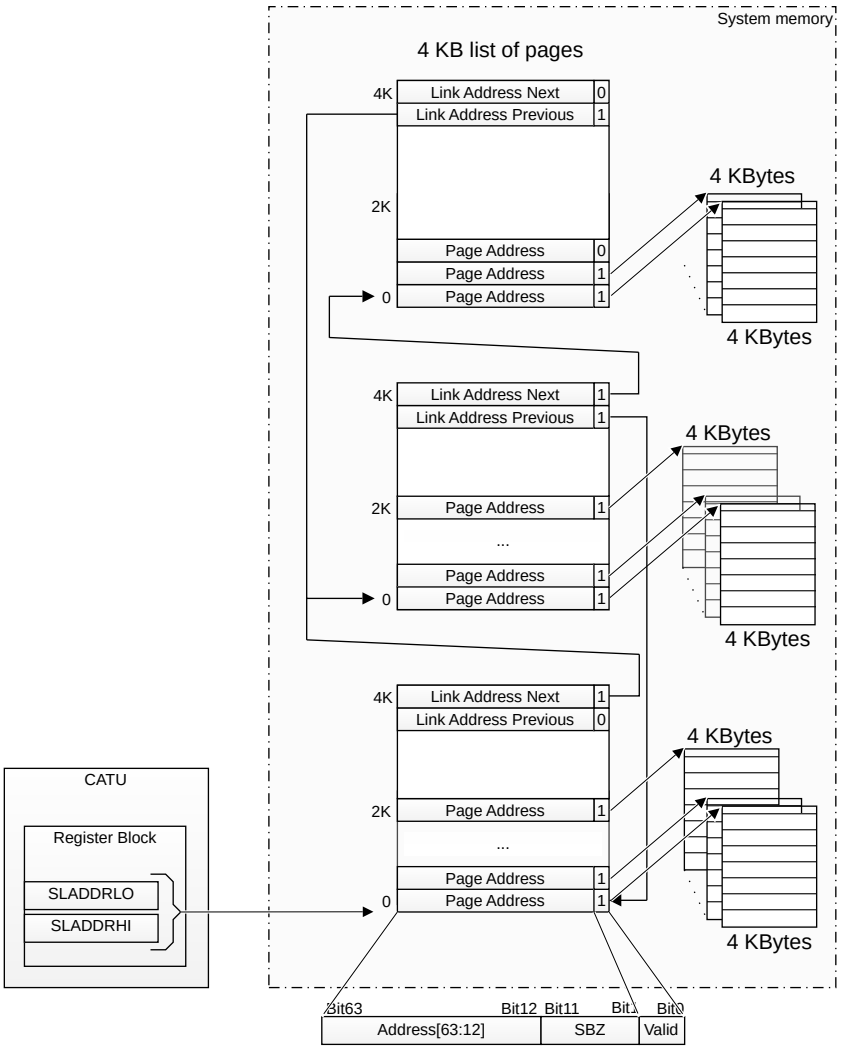
2. Bits[19:12] of the VA are used as an index into the selected 4KB list to locate the 4KB page physical address.
3. The page physical address is extracted from the list and bits[11:1] of the PA are replaced with the same bits from the VA.
4. Bit[0] of the PA indicates whether this page address is valid.

Note

You must program the valid bit, bit[0], in the PAs for all addresses, including the addresses of the previous and next link lists that are contained in each of the top 2KB subpages. If a scatter list walk reads in a scatter list entry that does not have the valid bit set, then it is assumed that the top address page has been reached.

The following figure shows the overall structure of the scatter list.

Figure 4-22: Scatter list structure



4.10.7.2 Programming the scatter list walker

The scatter list walker receives translation requests from the translation control block and initiates an AXI transaction to read entries from the scatter list. It generates translation responses after the AXI read response is received.

The scatter list walker is initialized when the CATU is enabled in translate mode. During initialization, internal Translation Lookaside Buffers (TLBs) are populated by fetching the translation values stored at SLADDRHI and SLADDRLO over the AXI manager interface. TLBs cache recently accessed address translations.

The scatter list forms a double-linked list. This double-linked list ensures that the list itself is not larger than 4KB, and that the list can be walked through in both directions.

The translation control block is responsible for controlling the address translation. In the event of a failed lookup in the TLB, translation requests are sent to the scatter list walker. There are two types of translation request:

TLB update

In the case of a write address, the scatter list walker generates a read request for the current and the next page address. After receiving the read response, it generates a translation response and the WR and WR_NEXT TLB entries are filled. In the case of a read address, the scatter list walker generates a read request for the current page address. After receiving the read response, it generates a translation response and the RD TLB entry is filled.

TLB prefetch

In this case, the scatter list walker generates a read request for the next page only. After receiving the read response, the WR_NEXT TLB entry is updated. The AXI AR channel cache and prot signals are set according to the AXICTRL register.

4.10.7.3 Address validation

The input address registers define the lower boundary of the valid address range.

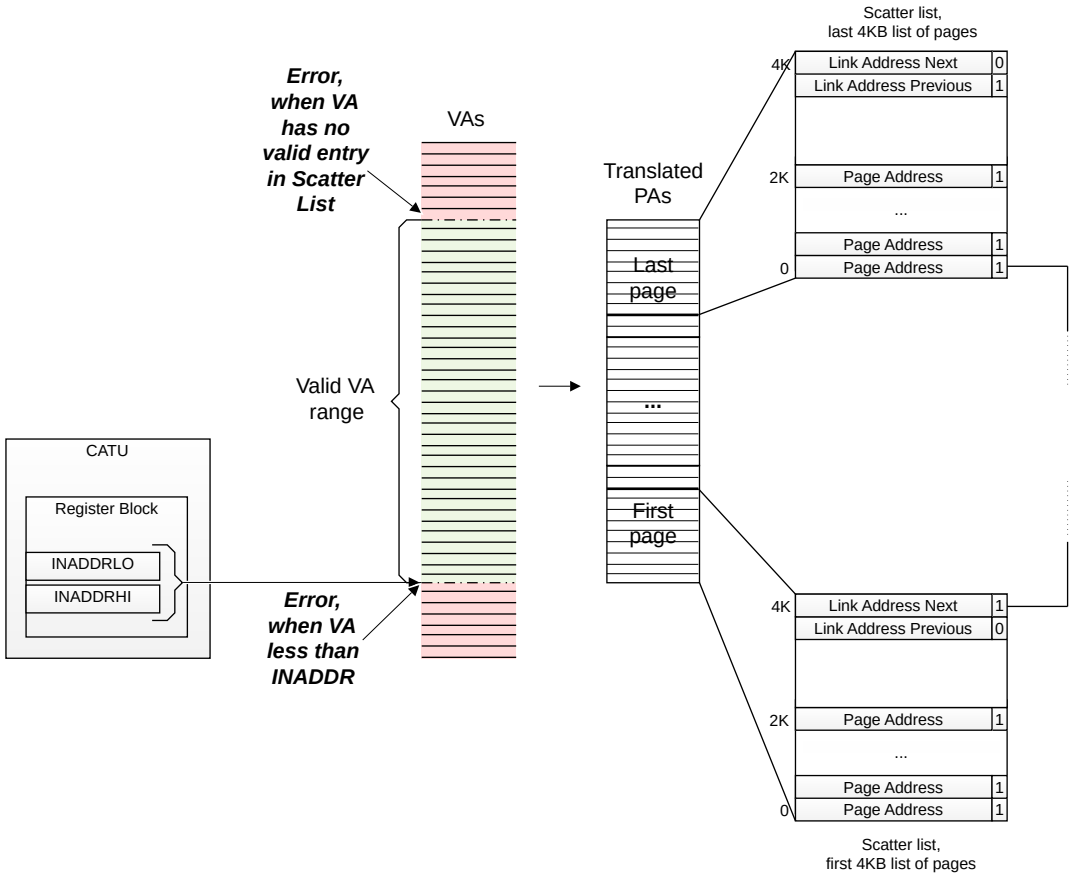
The input address registers are SLADDRHI and SLADDRLO.

If the VA is less than the address that is defined in the input address registers, then the CATU sets the STATUS.ADDRERR bit. If the interrupt is enabled, the CATU also generates an interrupt to indicate an address error.

The scatter list walker iterates through the table until the correct translation is found or an invalid entry is hit. When the scatter list walker returns an entry with valid == 0, then the upper bound has been reached. In this case, if enabled, an interrupt is generated, the CATU sets the STATUS.ADDRERR bit, and the out-of-range incoming transaction gets an error response.

The following figure shows the address validation.

Figure 4-23: Address validation



4.10.8 Initializing the CATU

The CATU operates when it is initialized and enabled.

To initialize the CATU:

1. Set CONTROL.ENABLE = 0.
2. Wait until STATUS.READY == 1.
3. Initialize the scatter list in system memory. This step is not required if the CATU is in Bypass mode. For more information, see Scatter list.
4. Set up the CATU registers that the following table shows.
5. Set CONTROL.ENABLE = 1.

The following table shows the registers that must be initialized before the CATU is enabled.

Table 4-7: Registers that must be initialized

Address	Register	Description
0x004	MODE	Mode register

Address	Register	Description
0x008	AXICTRL	AXI control register
0x00C	IRQEN	Interrupt enable register
0x020	SLADDRLO	Scatter list address low register
0x024	SLADDRHI	Scatter list address high register
0x028	INADDRLO	Input address low register
0x02C	INADDRHI	Input address high register

4.10.9 Reprogramming the CATU

It might be necessary to reprogram the CATU.

To reprogram the CATU:

1. Set CONTROL.ENABLE = 0 to disable the CATU.
2. Wait until STATUS.READY == 1.
3. Change the scatter list in system memory, if necessary.
4. Reconfigure the registers that are shown in Initializing the CATU.
5. Set CONTROL.ENABLE = 1 to enable the CATU.

When CONTROL.ENABLE is changed from 0 to 1:

- The scatter list walker is initialized.
- The TLB is initialized.
- When the CATU is in Translate mode, STATUS.AXIERR and STATUS.ADDRERR are cleared.

When CONTROL.ENABLE is changed from 1 to 0:

- The addrerr interrupt signal is cleared.
- The AXI subordinate interface returns an error response for any incoming transaction.
- The address pipeline is drained.
- The CATU waits for all outstanding AXI transactions on the AXI manager interface to complete, but ignores all received responses.
- The TLB is cleared.
- STATUS.READY is set to 1 when the following are both true:
  - There are no outstanding AXI transactions.
  - There are no ongoing translations, translation requests, or TLB updates.

Note

The AXI subordinate interface returns an error response for any incoming transaction until CONTROL.ENABLE == 1.

4.10.10 Error handling

The CATU can generate two types of error: VA and AXI.

Both of these errors are visible in STATUS.ADDRERR and STATUS.AXIERR. A VA error also results in an interrupt assertion when IRQEN.IRQEN is set.

When an error is discovered, either by the processor receiving an interrupt or reading the STATUS register shows an error, follow the steps in Reprogramming the CATU.

4.10.11 Unpredictable behavior

Certain scenarios result in unpredictable behavior.

The scenarios are:

- Not writing the MODE, AXICTRL, IRQEN, SLADDRLO, SLADDRHI, INADDRLO, and INADDRHI control registers before setting CONTROL.ENABLE
- Writing an invalid value to any of the AXICTRL, SLADDRLO, SLADDRHI, INADDRLO, or INADDRHI registers
- Not correctly setting up the scatter list before setting CONTROL.ENABLE

5. Timestamp components functional description

The SoC-600 timestamp components are the timestamp generator, the timestamp replicator, and the timestamp interpolator.

For information about the functionality of each timestamp component, see:

- Timestamp generator
- Timestamp replicator
- Timestamp interpolator

5.1 Timestamp generator

The `css600_tsgen` timestamp generator generates a 64-bit rolling time for distribution to trace generating components, such as ETMs.

The timestamp generator has two APB interfaces:

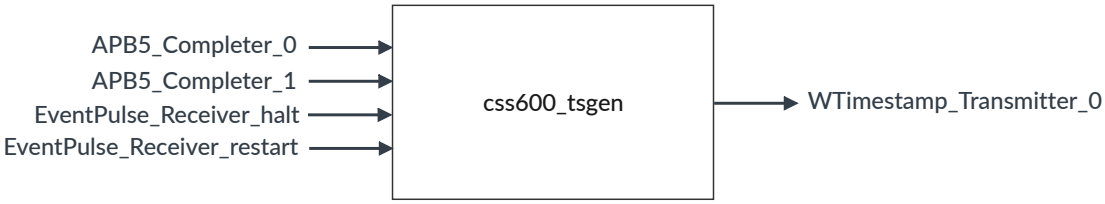
- A read-only interface, for reading the counter value.
- A programming interface. The system must ensure that this interface is accessible only to Secure software.

The counter in the timestamp generator also has the following key features:

- It runs at a constant clock frequency, regardless of the power and clocking state of the processor cores that are using it.
- When enabled and running, it can increment by 1 only.
- The counter continues to run in all levels of power down, other than when the system is turned off.
- The counter starts from 0.
- The counter value can be read using a 32-bit read on an APB interface.
- The counter value can be written only when it is either halted or disabled.
- When the system is halted as a result of debug, the counter can be programmed to either halt or continue incrementing.

The following figure shows the external connections on the timestamp generator.

Figure 5-1: css600_tsgen logical connections



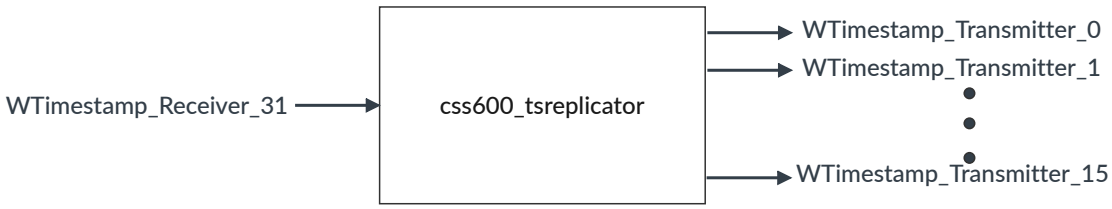
5.2 Timestamp replicator

The `css600_tsreplicator` is an IP-XACT phantom component that supports stitching in an IP-XACT tooling product. There is no Verilog module for `css600_tsreplicator`.

Use the `css600_tsreplicator` to connect a single Wide Timestamp (WTS) transmitter interface to multiple WTS receiver interfaces.

The following figure shows the external connections on the timestamp replicator.

Figure 5-2: css600_tsreplicator logical connections



5.3 Timestamp interpolator

The timestamp interpolator increases the resolution of a timestamp.

The interpolator shifts the input timestamp left by 8 bits, and uses the extra low-order bits to provide a more accurate timestamp value. The greater accuracy is achieved by monitoring changes to the input timestamp value over time to predict how fast it counts.

The following figure shows the external connections on the timestamp interpolator.

Figure 5-3: css600_tsintp logical connections

5.3.1 Functional interface

The timestamp interpolator adjusts to changes in the rate of the incoming timestamp.

The interpolator ensures that the interpolated timestamp never counts backwards, and pauses incrementing the interpolated timestamp if it gets ahead of the input timestamp value.

5.3.2 Low-Power Interface

The timestamp interpolator has a Low-Power Interface (LPI) to manage power reduction using high-level clock gating.

If the clock to the interpolator must be gated off, then the clock controller must use the LPI. When the interpolator exits the low-power state, it automatically recalculates the interpolation ratio before advancing the interpolated timestamp.

5.3.3 Limitations

Use of the timestamp interpolator is subject to some limitations.

The limitations are:

- The timestamp interpolator must not be used in the timestamp network that distributes processor time.
- There must be only one timestamp interpolator between the timestamp generator and a component that receives the timestamp.

6. Embedded Cross Trigger components functional description

The Embedded Cross Trigger (ECT) components enable CoreSight SoC-600 components to broadcast events between each other.

Events are distributed as follows:

- Each event type is connected to a trigger input on a Cross Trigger Interface (CTI).
- Each CTI can be programmed to connect each trigger input to each of 4 (default) or 16 channels. If programmed to do so, it causes an event on the corresponding channel when an input event occurs.

CTIs are connected to each other using one or more Cross Trigger Matrices (CTMs), through channel interfaces. When an event occurs on a channel, it is broadcast on that channel to all other CTIs in the system.

Each CTI can be programmed to connect each channel to each of several trigger outputs. If programmed to do so, it causes an event on the trigger output when a channel event occurs. Each CTI trigger output can be connected to a CoreSight component event input.

Cross triggering can take place between trigger inputs and outputs on a single CTI, or between multiple CTIs. CTIs can be programmed not to broadcast events for selected channels, so that certain events can only trigger output events on the same CTI. Only the CTIs are programmable, not the CTMs.

6.1 Event signaling protocol

The cross-triggering system does not attempt to interpret the events that are signaled through it. Events between CTI components and debug system components are transmitted using one of three mechanisms.

For CTI input events:

1. The event is signaled as a single-cycle clock pulse - an EventPulse. This option is selected using the CTI event configuration option `EVENT_IN_LEVEL` = 0.
2. The event is signaled from the debug components as a request-acknowledge signal pair and signaled to the CTI input as a single-cycle EventPulse. This option is selected using the CTI event configuration option `EVENT_IN_LEVEL` = 0. Connecting the debug component eventrequest-acknowledge signals to the CTI event input using the transmitter side of an Event Pulse asynchronous bridge.
3. The event is signaled from the debug component as a level sensitive event - an EventLevel. This option is selected using the CTI event configuration option `EVENT_IN_LEVEL` = 1.

For CTI output events:

1. The event is signaled as a single-cycle clock pulse - an EventPulse. This option is selected using the CTI event configuration option `SW_HANDSHAKE` = 0.
2. The event is signaled to the debug components as a request-acknowledge signal pair and signaled from the CTI output as a single-cycle EventPulse. To select this option:
   - Use the CTI event configuration option `SW_HANDSHAKE` = 0.
   - Connect the debug component event request-acknowledge signals to the CTI event output using the receiver side of an Event Pulse asynchronous bridge.
3. The event is signaled to the debug component as a level sensitive event - an EventLevel. This option is selected using the CTI event configuration option `SW_HANDSHAKE` = 1.

Events are broadcast between CTI and CTM components on the cross-trigger channels as a pulse. When an event passes across a clock domain boundary using an asynchronous bridge, handshaking occurs to ensure that the event lasts for exactly one clock cycle in the destination clock domain.

Each channel is a shared broadcast medium that can carry events from multiple sources going to multiple domains. When a CTI sends events onto a channel, they can coincide with other events on the same channel, so that the events become pulses of more than one clock cycle. This behavior is normal within the cross trigger system.

In usage models that count events that are passed through the cross-triggering system, events that occur close together might be merged into a single event with a single pulse when passed to another clock domain.

6.2 Cross Trigger Interface

The SoC-600 `css600_cti` component connects one or more event sources and one or more event destinations to the cross trigger network.

The CTI has the following functional interfaces:

- Up to 32 trigger inputs, enabling events to be signaled to the CTI.
- Up to 32 trigger outputs, enabling the CTI to signal events to other components.
- A channel interface for connecting CTIs together using one or more CTMs.
- An APB interface for accessing the registers of the CTI.
- An Authentication interface for controlling access to certain debug events.
- Eight asicctrl signals that can be used to control external multiplexers.

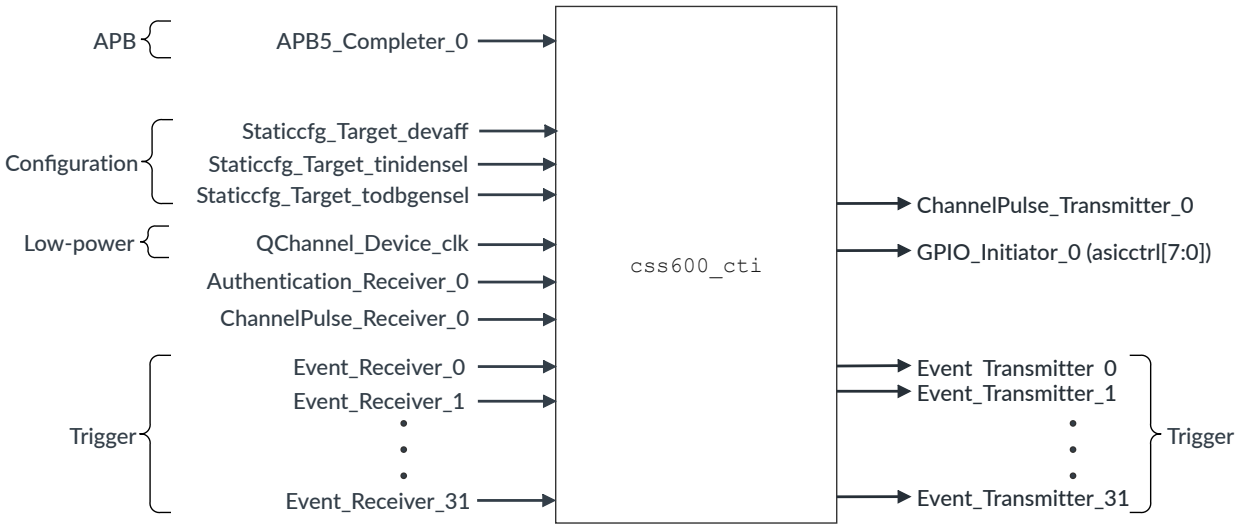
The CTI includes configuration tie-off inputs that enable several trigger input and output types to be connected.

We recommend that the CTI that is connected to a processor is disabled before the processor clock is stopped. This operation minimizes the likelihood of unexpected events entering the cross triggering system or affecting the processor when its clock is restarted. The CTICONTROL.CTIEN

register bit can be used to disable the CTI globally without changing the event mapping programming.

The following figure shows the external connections on the Cross Trigger Interface.

Figure 6-1: css600_cti logical connections



6.2.1 asicctrl

The asicctrl[7:0] output signals of the CTI can be used to control multiplexing on a CTI event input if necessary.

The exact configuration of any external multiplexing is user-defined. We do not define any relationship between values that are written to the control register and the actual configuration of the multiplexers.

The system integrator sets the `EXT_MUX_NUM` parameter to indicate the configuration of any external multiplexers. We do not specify the usage of the `EXT_MUX_NUM` parameter. See your system integrator for details of the implementation of your specific SoC.

6.3 Cross Trigger Matrix

The `css600_ctm` connects CTI components together in a cross trigger system.

The component is configurable for up to 33 channel interfaces. If more than 33 CTIs must be connected together, then CTMs can be connected together without limitation. Each channel interface can support either 4 or 16 channels.

The following figure shows the external connections on the Cross Trigger Matrix.

Figure 6-2: css600_ctm logical connections

6.4 Event Pulse to Event adapter

The `css600_eventpulsetoeventadapter` Event Pulse to Event adapter is a wrapper component that instantiates a receiver half of a pulse async bridge with a configurable signal width.

The component provides the req/ack handshake that is required to interface a SoC-600 event to a legacy CTI, such as one in a CoreSight™ SoC-400 system.

The following figure shows the external connections on the Event Pulse to Event adapter.

Figure 6-3: css600_eventpulsetoeventadapter logical connections

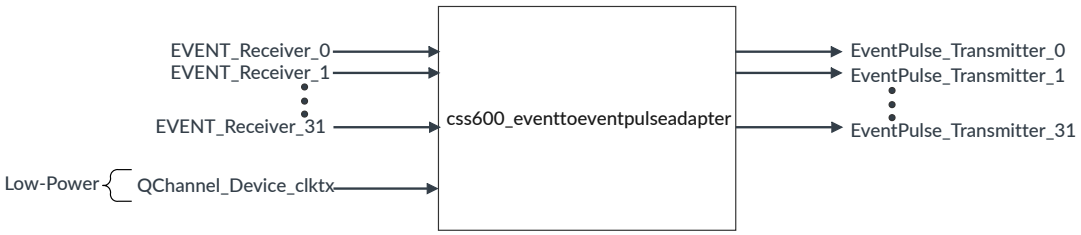
6.5 Event to Event Pulse adapter

The `css600_eventtoeventpulseadapter` Event to Event Pulse adapter is a wrapper component that instantiates a transmitter half of a pulse async bridge with a configurable signal width.

The component provides the req/ack handshake that is required to interface a legacy event source to a SoC-600 CTI.

The following figure shows the external connections on the Event to Event Pulse adapter.

Figure 6-4: css600_eventtoeventpulseadapter logical connections



6.6 Event Level asynchronous bridge

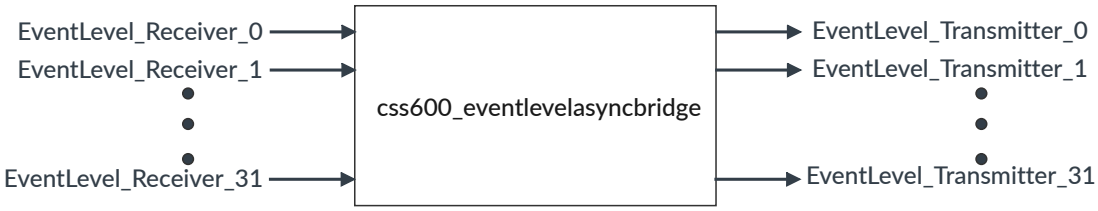
The `css600_eventlevelasyncbridge` Event Level asynchronous bridge is a wrapper component that instantiates a synchronizer.

The bridge passes an event that operates as a level, rather than a pulse, across a clock domain boundary. The bridge can be used, for example, when using the software handshake configuration of a CTI event output, and the resulting event output must cross a clock or power domain boundary to reach its destination.

If more than one signal is to be transported across the same boundary, then the component can be configured for width.

The following figure shows the external connections on the Event Level asynchronous bridge.

Figure 6-5: css600_eventlevelasyncbridge logical connections



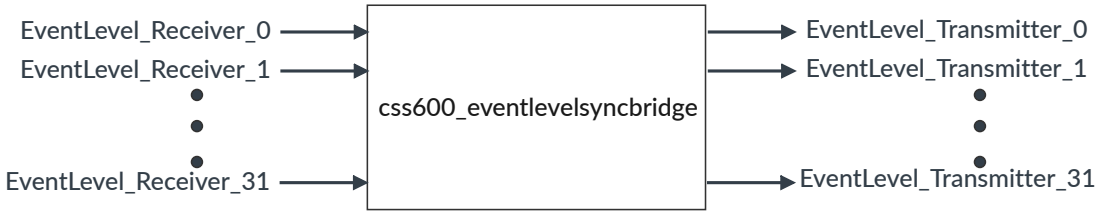
6.7 Event Level synchronous bridge

The `css600_eventlevelsyncbridge` Event Level synchronous bridge is a wrapper component that instantiates a register slice. Use a register slice as an aid to timing closure on events that must travel a long distance on chip.

If more than one signal is to be transported across the same boundary, then the component is configurable for width.

The following figure shows the external connections on the Event Level synchronous bridge.

Figure 6-6: css600_eventlevelsyncbridge logical connections



6.8 Event Pulse asynchronous bridge

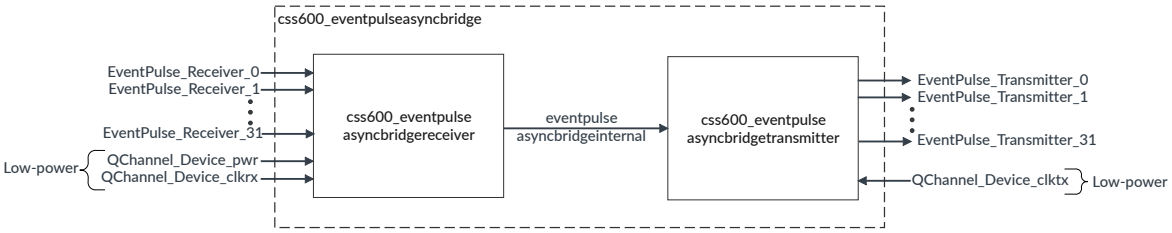
The `css600_eventpulseasyncbridge` Event Pulse asynchronous bridge is used where an event signal, or a group of events, must cross a clock or power domain boundary.

The bridge has the following features:

- Two independent clock domains with any phase or frequency alignment
- Two independent power domains, either of which can be switched relative to the other
- Three Q-Channel LPIs for receiver side clock, transmitter side clock, and power switching management
- Two-part meta-component with separate receiver and transmitter side components
- Configurable up to 32-bits wide for transporting multiple events across the same boundary

The following figure shows the external connections on the Event Pulse asynchronous bridge.

Figure 6-7: css600_eventpulseasyncbridge logical connections



6.9 Event Pulse synchronous bridge

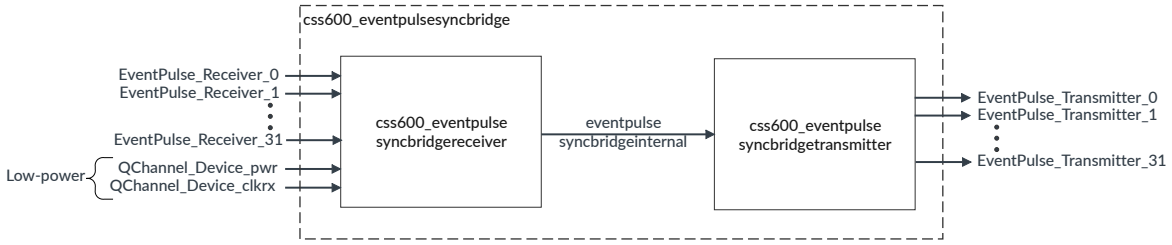
The `css600_eventpulsesyncbridge` Event Pulse synchronous bridge is used where an event signal, or a group of events, must cross a synchronous clock domain boundary.

The bridge has the following features:

- Two synchronous clock domains with any frequency difference. The clocks must be skew balanced and from a common source so that they are high-level-gated by a common control point.
- Two Q-Channel LPIs for clock and power switching management
- Two-part meta-component with separate receiver and transmitter side components
- Configurable up to 32 bits wide for transporting multiple events across the same boundary

The following figure shows the external connections on the Event Pulse synchronous bridge.

Figure 6-8: css600_eventpulsesyncbridge logical connections



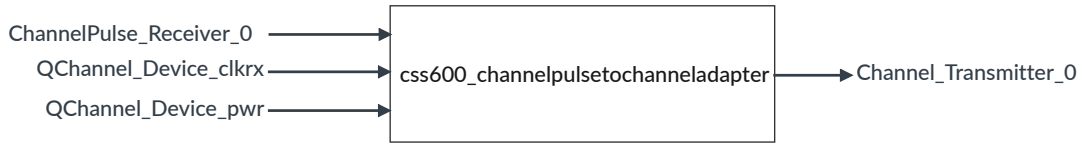
6.10 Channel Pulse to Channel adapter

The `css600_channelpulsetochanneladapter` Channel Pulse to Channel adapter is a wrapper component that instantiates a receiver half of a pulse async bridge.

The component provides the req/ack handshake that is required to interface a SoC-600 CTI or CTM to a legacy CTI or CTM such as one in a CoreSight™ SoC-400 system.

The following figure shows the external connections on the Channel Pulse to Channel adapter.

Figure 6-9: css600_channelpulsetochanneladapter logical connections



6.11 Channel to Channel Pulse adapter

The `css600_channeltochannelpulseadapter` Channel to Channel Pulse adapter is a wrapper component that instantiates a transmitter half of a pulse async bridge.

The component provides the req/ack handshake that is required to interface a SoC-600 CTI or CTM to a legacy CTI or CTM such as one in a CoreSight™ SoC-400 system.

The following figure shows the external connections on the Channel to Channel Pulse adapter.

Figure 6-10: css600_channeltochannelpulseadapter logical connections

6.12 Channel Pulse asynchronous bridge

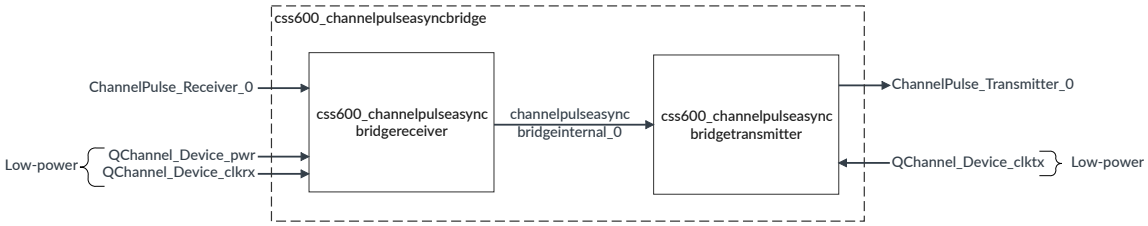
The `css600_channelpulseasyncbridge` Channel Pulse asynchronous bridge is a wrapper component that instantiates a pulse asynchronous bridge.

The bridge connects a CTI to a CTM, or two CTMs, where the signals must cross a clock or power domain boundary. The bridge has the following features:

- Two independent clock domains with any phase or frequency alignment
- Two independent power domains, either of which can be switched relative to the other
- Two-part meta-component with separate receiver and transmitter side components

The following figure shows the external connections on the Channel Pulse asynchronous bridge.

Figure 6-11: css600_channelpulseasyncbridge logical connections



6.13 Channel Pulse synchronous bridge

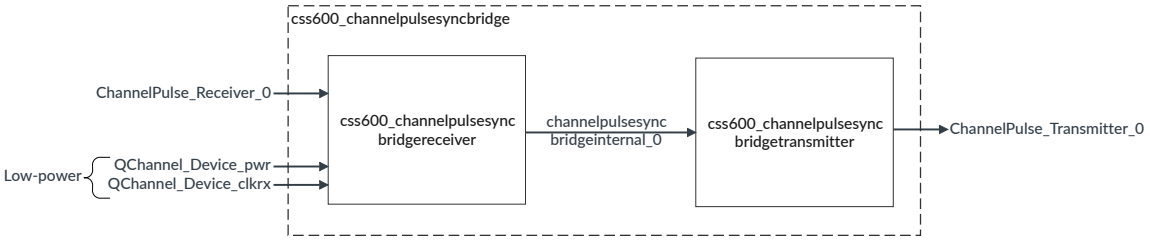
The `css600_channelpulsesyncbridge` Channel Pulse synchronous bridge is a wrapper component that instantiates a pulse synchronous bridge.

The bridge connects a CTI to a CTM, or two CTMs, where the signals must cross a synchronous clock domain boundary. The bridge has the following features:

- Two synchronous clock domains with any frequency difference. The clocks must be skew balanced, and from a common source, so that they are high-level-gated by a common control point.
- Two Q-Channel LPIs for clock and power switching management.
- Two-part meta-component with separate receiver and transmitter side components.

The following figure shows the external connections on the Channel Pulse synchronous bridge.

Figure 6-12: css600_channelpulsesyncbridge logical connections



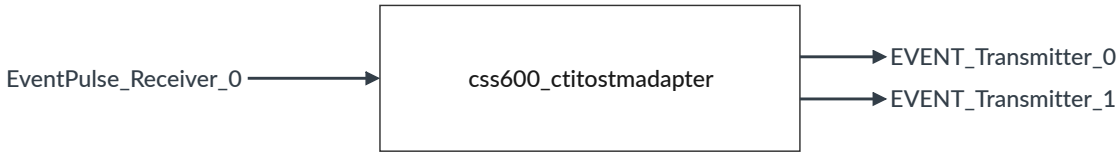
6.14 CTI to STM adapter

The `css600_ctitostmadapter` is an IP-XACT phantom component that supports stitching in an IP-XACT tooling product. There is no Verilog module for `css600_ctitostmadapter`.

The `css600_ctitostmadapter` CTI to STM adapter adapts a single event signal to two event inputs of a System Trace Macrocell.

The following figure shows the external connections on the CTI to STM adapter.

Figure 6-13: css600_ctitostmadapter logical connections



7. Authentication components functional description

The SoC-600 authentication components are the authentication replication, the authentication asynchronous bridge, and the authentication synchronous bridge.

- Authentication replicator
- Authentication asynchronous bridge
- Authentication synchronous bridge

7.1 Authentication replicator

The `css600_authreplicator` is an IP-XACT phantom component that is supports stitching in an IP-XACT tooling product.

There is no Verilog module for `css600_authreplicator`.

Use the `css600_authreplicator` to connect a single Authentication transmitter interface to multiple Authentication receiver interfaces.

The following figure shows the external connections on the Authentication replicator.

Figure 7-1: css600_authreplicator logical connections

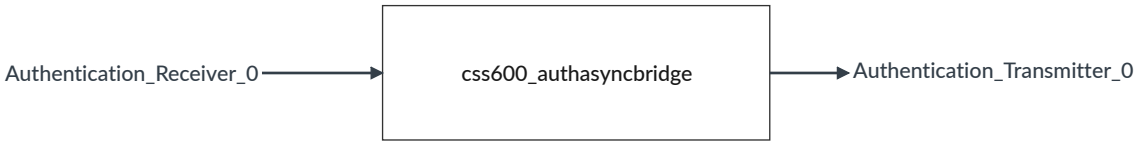
7.2 Authentication asynchronous bridge

The `css600_authasyncbridge` authentication asynchronous bridge is used where the Authentication interface must cross a clock or power domain boundary.

The bridge contains synchronizers to capture the signals in the receiving clock domain.

The following figure shows the external connections on the Authentication asynchronous bridge.

Figure 7-2: css600_authasyncbridge logical connections

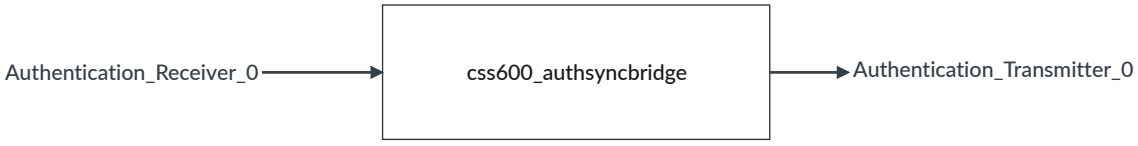


7.3 Authentication synchronous bridge

The `css600_authsyncbridge` authentication synchronous bridge is a register slice to aid timing closure for authentication signals that are crossing a large distance across a chip.

The following figure shows the external connections on the Authentication synchronous bridge.

Figure 7-3: css600_authsyncbridge logical connections



8. Processor Integration Layer components

The Cortex® Processor Integration Layers (PILs) allow component integration with processors. Each processor has its own PIL. Each PIL also has interfaces that are specific to the corresponding processor.


Note

The PILs in this document are delivered with the CoreSight SoC-600 components. For integration information for other processors, see the corresponding processor documentation.

8.1 Cortex-A5 PIL overview

The Cortex®-A5 Processor Integration Layer (PIL) provides a configurable example integration of the processor and several tightly coupled debug components.

The Cortex-A5 PIL is configurable and consists of the following:

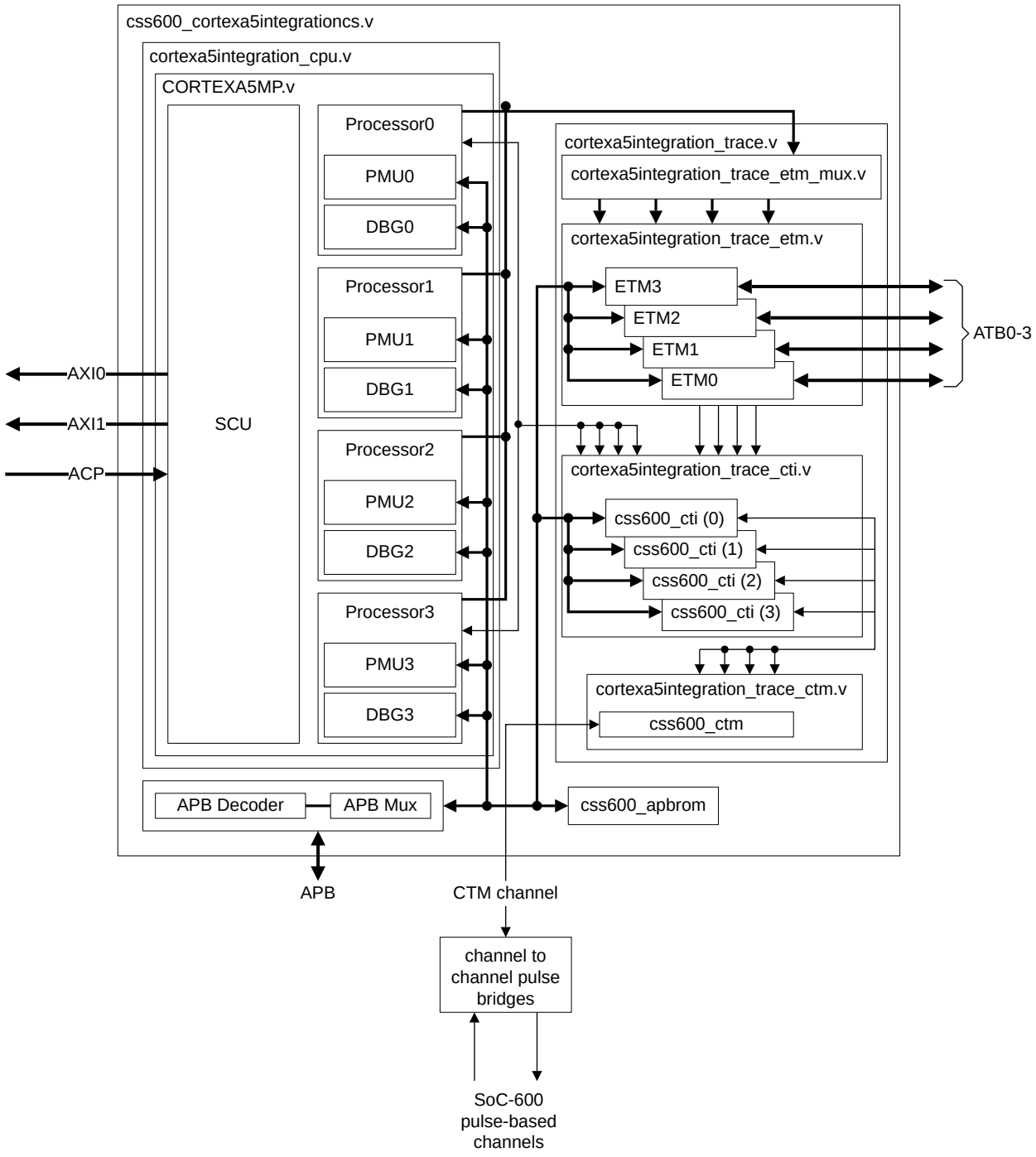
- A ROM table
- An APB subsystem
- A Cross Trigger Matrix (CTM)
- Logic that enables sharing an ETM trace unit between several processors
- Zero to four Cross Trigger Interfaces (CTIs), referred to as CTI0, CTI1, CTI2, and CTI3
- Zero to four Embedded Trace Macrocell (ETM) trace units, referred to as ETM0, ETM1, ETM2, and ETM3
- A Cortex-A5 uniprocessor, referred to as Processor0, or a Cortex-A5 MPCore processor with up to four processors, referred to as Processor0, Processor1, Processor2, and Processor3

The Cortex-A5 PIL has the following interfaces:

- APB debug
- An Embedded Cross Trigger (ECT) channel interface
- Zero to four ATB trace outputs
- One or two AXI interfaces for connection to the memory system
- Accelerator Coherency Port (ACP) to provide memory coherency between each processor in the Cortex-A5 PIL and an external requester

The following figure shows a block diagram of the Cortex-A5 PIL.

Figure 8-1: Cortex-A5 PIL block diagram



8.1.1 Cortex-A5 PIL CoreSight component identification

CoreSight™ components have several IDs that identify the components.

The following table shows the CoreSight ID register reset values for the components present within the Cortex®-A5 PIL.

Table 8-1: Cortex-A5 PIL CoreSight ID register reset values

PID	CID	DevType	DevArch	Revision	Component
0x00000004002BB4A5	0xB105900D	0x00	0x47700AF7	r0p1	css600_cortexa5integrationcs ROM table
0x00000004002BB955	0xB105900D	0x13	0x00000000	r0p2	Cortex-A5 ETM
0x00000004004BB9ED	0xB105900D	0x14	0x47701A14	r1p0	css600_cti
0x00000004001BB9A5	0xB105900D	0x16	0x00000000	r0p1	Cortex-A5 PMU
0x00000004001BBC05	0xB105900D	0x15	0x00000000	r0p1	Cortex-A5 Debug

See the *Arm® CoreSight™ Architecture Specification v3.0* for information on the CoreSight ID scheme.

8.1.2 Cortex-A5 PIL Debug memory map

The debug components in the Cortex®-A5 PIL share memory space with the processor system.

The paddrdbg[31] signal is inverted and mapped to the paddrdbg[17] signal inside the PIL. The following table shows the locations of the Cortex-A5 PIL CoreSight™ components.

See the *Arm® CoreSight™ Architecture Specification v3.0* for information on the CoreSight ID scheme.

Table 8-2: Cortex-A5 PIL debug memory map

APB address range	Components	Comments
0x00000000-0x00000FFF	ROM table	Start of external view of debug memory space
...	...	...
0x00010000-0x00010FFF	Processor 0 debug components	The processor internally uses the paddrdbg[12] signal to separate debug from the PMU
0x00012000-0x00012FFF	Processor 0 PMU	-
0x00013000-0x00013FFF	Processor 1 debug components	-
...	...	...
0x00017000-0x00017FFF	Processor 3 PMU	-
0x00018000-0x00018FFF	CTI0	There is one CTI for each processor
...	...	...
0x0001B000-0x0001BFFF	CTI3	-
0x0001C000-0x0001CFFF	ETM0	There is one ETM for each processor, or one shared ETM for all processors
...	...	...
0x0001F000-0x0001FFFF	ETM3	-
0x00020000-0x00020FFF	Internal view of ROM table	Start of internal debug view of debug memory space

APB address range	Components	Comments
0x00030000-0x00030FFF	Processor 0 debug components	Location for self hosted debug access to processor debug components
0x00031000-0x00031FFF	Processor 0 PMU	Internal view
...	...	...
0x0003F000-0x0003FFFF	ETM3	Internal view

8.2 Cortex-A8 PIL overview

The Cortex®-A8 Processor Integration Layer (PIL) provides integration with trace and debug components and includes suitable interfaces.

The Cortex-A8 PIL consists of the following:

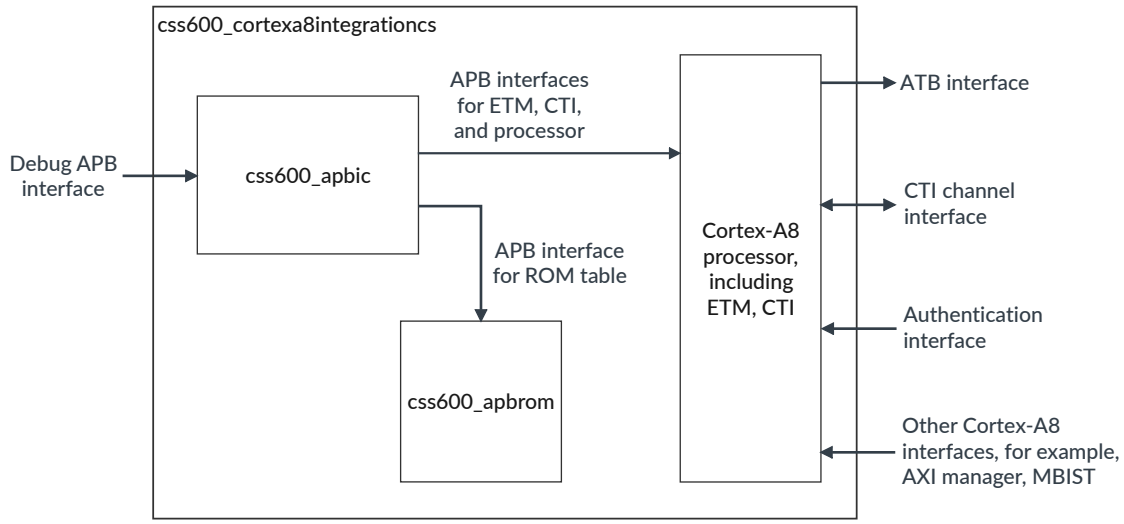
- An Embedded Trace Macrocell (ETM)
- A Cross Trigger Interface (CTI)
- A ROM table that enables automatic detection of the debug APB memory map
- Various processor debug components

The Cortex-A8 PIL has the following interfaces:

- Debug APB
- ATB trace
- CTI channel
- Authentication

The following figure shows a block diagram of the Cortex-A8 PIL.

Figure 8-2: Cortex-A8 PIL block diagram



8.2.1 Cortex-A8 PIL CoreSight component identification

CoreSight™ components have several IDs that identify the components.

The following table shows the CoreSight ID register reset values for the components present within the Cortex®-A8 PIL.

Table 8-3: Cortex-A8 PIL CoreSight ID register reset values

PID	CID	DevType	DevArch	Revision	Component
0x00000004001BB4A8	0xB105900D	0x00	0x47700AF7	r0p1	`css600_cortexa8integrationcs` ROM table
0x00000004206BB921	0xB105900D	0x13	0x00000000	r0p0	Cortex-A8 ETM
0x00000004003BB9ED	0xB105900D	0x14	0x47701A14	r0p3	Cortex-A8 CTI
0x00000004206BBC08	0xB105900D	0x15	0x00000000	r0p2	Cortex-A8 Debug

See the *Arm® CoreSight™ Architecture Specification v3.0* for information on the CoreSight ID scheme.

8.2.2 Cortex-A8 PIL Debug memory map

The debug components in the Cortex®-A8 PIL share memory space with the processor system.

The paddrdbg[31] signal is inverted and mapped to the paddrdbg[17] signal inside the PIL. The following tables show the locations of the Cortex-A8 PIL CoreSight™ components.

See the *Arm® CoreSight™ Architecture Specification v3.0* for information on the CoreSight ID scheme.

Table 8-4: Cortex-A8 PIL debug memory map

APB address range	Components	Comments
0x00000000-0x00000FFF	ROM table	Start of external view of debug memory space
...	...	...
0x00010000-0x00010FFF	Processor debug components	A single 4KB block is used in the Cortex-A8 processor
...	...	...
0x00018000-0x00018FFF	CTI	One CTI is present inside the processor
...	...	...
0x0001C000-0x0001CFFF	ETM	One optional ETM is present inside the processor
...	...	...
0x00020000-0x00020FFF	Internal view of ROM table	Start of internal debug view of debug memory space
0x00030000-0x00030FFF	Processor debug components	Location for self hosted debug access to processor debug components
...	...	...
0x0003C000-0x0003CFFF	ETM	Internal view

8.3 Cortex-A9 PIL overview

The Cortex®-A9 Processor Integration Layer (PIL) provides integration with trace and debug components and includes suitable interfaces.

The Cortex-A9 PIL consists of the following:

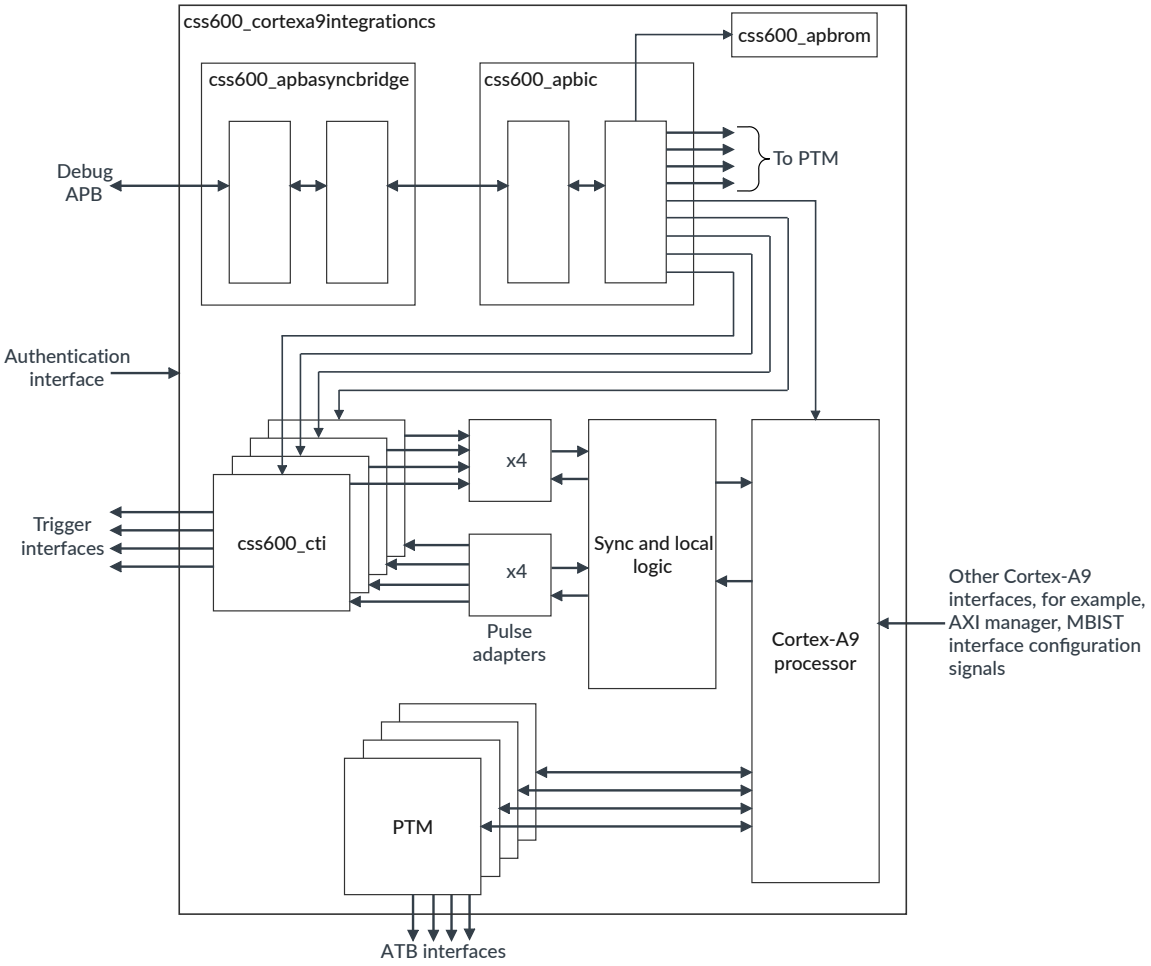
- Up to four Program Trace Macrocells (PTMs)
- Up to four Cross Trigger Interfaces (CTIs)
- A Cross Trigger Matrix (CTM)
- A ROM table that enables automatic detection of the debug APB memory map
- Various processor debug components

The Cortex-A9 PIL has the following interfaces:

- APB debug
- Authentication
- CTI channel
- Up to four ATB trace outputs
- Up to two AXI Snoop Control Unit (SCU) managers
- Up to two AXI processor managers
- AXI subordinate
- Design For Test (DFT)
- Timestamp
- Memory Built-in Self Test (MBIST)

The following figure shows a block diagram of the Cortex-A9 PIL.

Figure 8-3: Cortex-A9 PIL block diagram



8.3.1 Cortex-A9 PIL CoreSight component identification

CoreSight™ components have several IDs that identify the components.

The paddrdbg[31] signal is inverted and mapped to the paddrdbg[17] signal inside the PIL. The following table shows the CoreSight ID register reset values for the components present within the Cortex®-A9 PIL.

Table 8-5: Cortex-A9 PIL CoreSight ID register reset values

PID	CID	DevType	DevArch	Revision	Component
0x00000002001BB4A9	0xB105900D	0x00	0x47700AF7	r0p0	css600_cortexa9integrationcs ROM table

PID	CID	DevType	DevArch	Revision	Component
0x00000004001BB950	0xB105900D	0x13	0x00000000	r0p0	Cortex-A9 PTM
0x00000004003BB9ED	0xB105900D	0x14	0x47701A14	r0p3	css600_cti
0x00000004000BB9A0	0xB105900D	0x16	0x00000000	r4p1	Cortex-A9 PMU
0x00000004000BBC09	0xB105900D	0x15	0x00000000	r4p1	Cortex-A9 Debug

See the *Arm® CoreSight™ Architecture Specification v3.0* for information on the CoreSight ID scheme.

8.3.2 Cortex-A9 PIL Debug memory map

The debug components in the Cortex®-A9 PIL share memory space with the processor system.

The following table shows the locations of the Cortex-A9 PIL CoreSight™ components.

Table 8-6: Cortex-A9 PIL debug memory map

APB address range	Components	Comments
0x00000000-0x00000FFF	ROM table	Start of external view of debug memory space
...	...	...
0x00010000-0x000101FF, or 0x00010000-0x000103FF, or 0x00010000-0x000107FF	Processor 0 debug components	Depending on the multiprocessor configuration, the debug components occupy a memory space as follows: **Single processor** 13 bits, 8KB **Two processors** 14 bits, 16KB **Three or four processors** 15 bits, 32KB
...	...	...
0x00018000-0x00018FFF	CTI0	Always present
0x00019000-0x00019FFF	CTI1	Present if two or more processors are present
0x0001A000-0x0001AFFF	CTI1	Present if three or more processors are present
0x0001B000-0x0001BFFF	CTI3	Present if four processors are present.
0x0001C000-0x0001CFFF	PTM0	Present if PTM is configured.
0x0001D000-0x0001DFFF	PTM1	Present if PTM configured and processor 1 is present and no PTM sharing
0x0001E000-0x0001EFFF	PTM2	Present if PTM configured and processor 2 is present and no PTM sharing
0x0001F000-0x0001FFFF	PTM3	Present if PTM configured and processor 3 is present and no PTM sharing
0x00020000-0x00020FFF	Internal view of ROM table	Start of internal debug view of debug memory space
0x00030000-0x000301FF, or 0x00030000-0x000303FF, or 0x00030000-0x000307FF	Processor debug components	Location self hosted debug access to processor debug components. Same dependencies on configuration as external view.
...	...	...
0x0003F000-0x0003FFFF	PTM3	Location of self-hosted access to PTM3

8.4 Cortex-R4 PIL overview

The Cortex®-R4 Processor Integration Layer (PIL) provides integration with trace and debug components and includes suitable interfaces.

The Cortex-R4 PIL integrates the Cortex-R4 processor with:

- ETM-R4
- A Cross Trigger Interface (CTI)
- A ROM table that enables automatic detection of the debug APB memory map

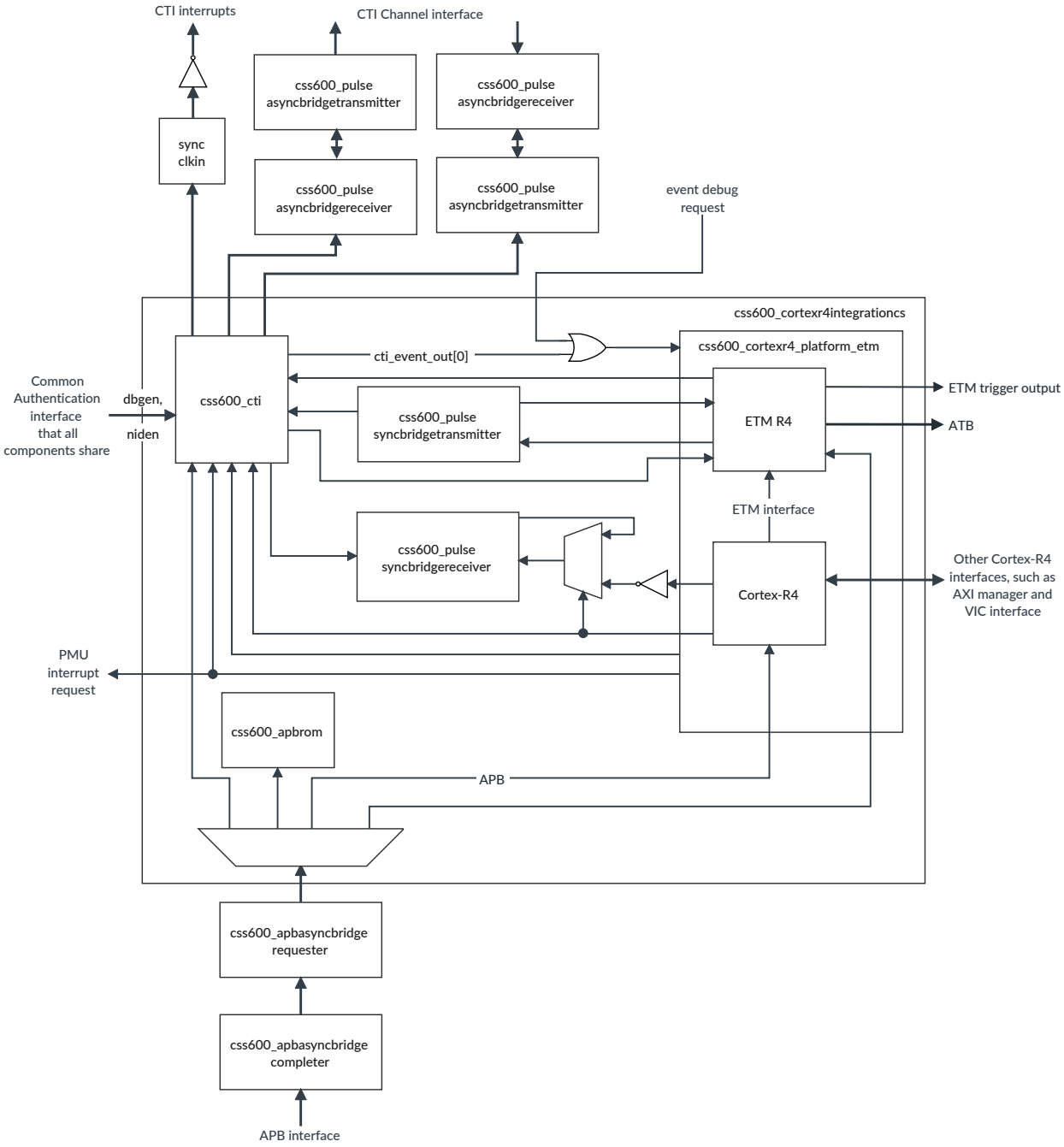
All of the non-debug interfaces of the processor are exposed at the top-level of the PIL. The address buses for the AXI subordinate interface are 32 bits wide on the Cortex-R4 PIL, but only 23 bits wide on the Cortex-R4 processor. The PIL does not use the top nine bits of these buses. Other non-debug interfaces are the same as on the processor. The processor documentation describes these interfaces.

The Cortex-R4 PIL has the following interfaces:

- AXI subordinate
- TCM ports
- AXI manager
- APB debug
- Authentication
- ATB trace output
- Configuration signals
- Test and MBIST signals
- VIC interface and interrupt signals
- Miscellaneous status and control

The PIL contains a single APB debug interface that can be asynchronous to the processor clock. It has one ATB trace output interface and an authentication interface. The following figure shows the Cortex-R4 PIL.

Figure 8-4: Cortex-R4 PIL block diagram



8.4.1 Cortex-R4 PIL CoreSight component identification

CoreSight™ components have several IDs that identify the components.

The following table shows the CoreSight ID register reset values for the components present within the Cortex®-R4 PIL.

Table 8-7: Cortex-R4 PIL CoreSight ID register reset values

PID	CID	DevType	DevArch	Revision	Component
0x00000004002BB4B4	0xB105900D	0x00	0x47700AF7	r1p0	css600_cortexr4integrationcs ROM table
0x00000004003BB930	0xB105900D	0x13	0x00000000	r0p0	Cortex-R4 ETM
0x00000004004BB9ED	0xB105900D	0x14	0x47701A14	r1p0	css600_cti
0x00000004000BBC14	0xB105900D	0x15	0x00000000	r1p4	Cortex-R4 Debug

See the *Arm® CoreSight™ Architecture Specification v3.0* for information on the CoreSight ID scheme.

8.4.2 Cortex-R4 PIL Debug memory map

The debug components in the Cortex®-R4 PIL share memory space with the processor system.

The following table shows the locations of the Cortex-R4 PIL CoreSight™ components.

Table 8-8: Cortex-R4 PIL debug memory map

APB address range, PADDRDBG[16:0]	Components
0x00000-0x00FFF	ROM table
0x10000-0x10FFF	Cortex-R4 processor
0x18000-0x18FFF	CTI
0x1C000-0x1CFFF	ETM-R4

8.5 Cortex-R5 PIL overview

The Cortex®-R5 Processor Integration Layer (PIL) integrates the Cortex-R5 processor. The Cortex-R5 processor might include one or two processors.

The PIL also integrates:

- Up to two Embedded Trace Macrocells (ETMs): one ETM-R5 for each processor, or a single ETMR5 shared between two processors, or no ETM-R5 at all
- One Cross Trigger Interface (CTI) for each processor
- A ROM table that enables automatic detection of the debug APB memory map

All of the non-debug interfaces of the processor are exposed at the top-level of the PIL. The processor documentation describes these interfaces.

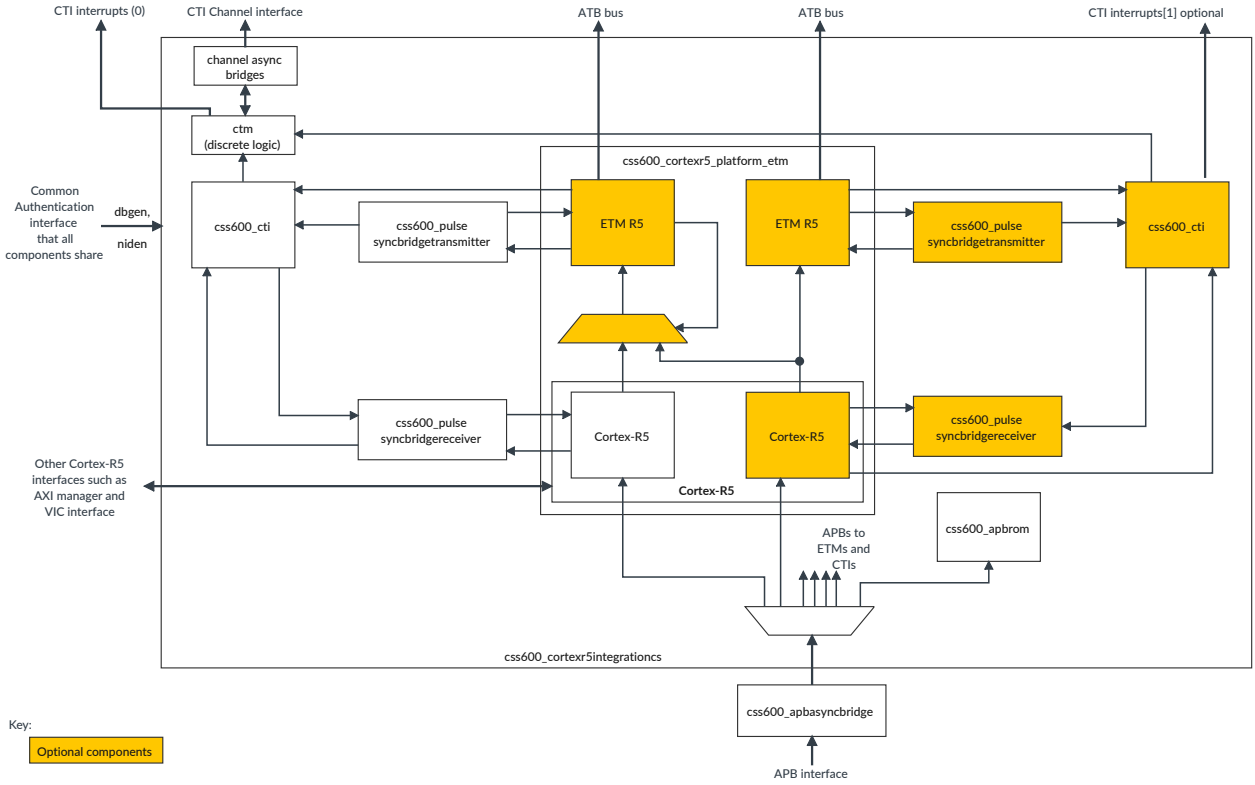
The Cortex-R5 PIL has the following interfaces:

- APB debug
- ATB trace output
- Authentication
- AXI manager

- AXI subordinate
- VIC interface and interrupt signals
- TCM ports
- Test and MBIST signals
- Configuration signals
- Miscellaneous status and control

The PIL contains a single APB debug interface that can be asynchronous to the processor clock. It has one ATB trace output interface and an authentication interface. The following figure shows the Cortex-R5 PIL.

Figure 8-5: Cortex-R5 PIL block diagram



8.5.1 Cortex-R5 PIL CoreSight component identification

CoreSight™ components have several IDs that identify the components.

The following table shows the CoreSight ID register reset values for the components present within the Cortex®-R5 PIL.

Table 8-9: Cortex-R5 PIL CoreSight ID register reset values

PID	CID	DevType	DevArch	Revision	Component
0x00000004002BB4B1	0xB105900D	0x00	0x47700AF7	r1p0	css600_cortexr5integrationcs ROM table
0x00000004000BB931	0xB105900D	0x13	0x00000000	r0p0	Cortex-R5 ETM
0x00000004004BB9ED	0xB105900D	0x14	0x47701A14	r1p0	css600_cti
0x00000004000BBC15	0xB105900D	0x15	0x00000000	r1p2	Cortex-R5 Debug

See the *Arm® CoreSight™ Architecture Specification v3.0* for information on the CoreSight ID scheme.

8.5.2 Cortex-R5 PIL Debug memory map

The debug components in the Cortex®-R5 PIL share memory space with the processor system.

The following table shows the locations of the Cortex-R5 PIL CoreSight™ components.

Table 8-10: Cortex-R5 PIL debug memory map

APB address range, PADDRDBG[16:0]	Components
0x00000-0x00FFF	ROM table
0x10000-0x10FFF	Cortex-R5 processor0
0x12000-0x12FFF	Cortex-R5 processor1, if present
0x18000-0x18FFF	CTI for processor0
0x19000-0x19FFF	CTI for processor1, if present
0x1C000-0x1CFFF	ETM for processor0, or shared ETM
0x1D000-0x1DFFF	ETM for processor1, if not shared

8.6 Cortex-M0 PIL overview

The Cortex®-M0 Processor Integration Layer (PIL) integrates the processor with trace and debug components and includes suitable interfaces.

The Cortex-M0 PIL consists of the following:

- A Cortex-M0 processor
- An optional Wakeup Interrupt Controller (WIC)
- A ROM table to identify the PIL contents
- A Cross Trigger Interface (CTI) for debug event communication

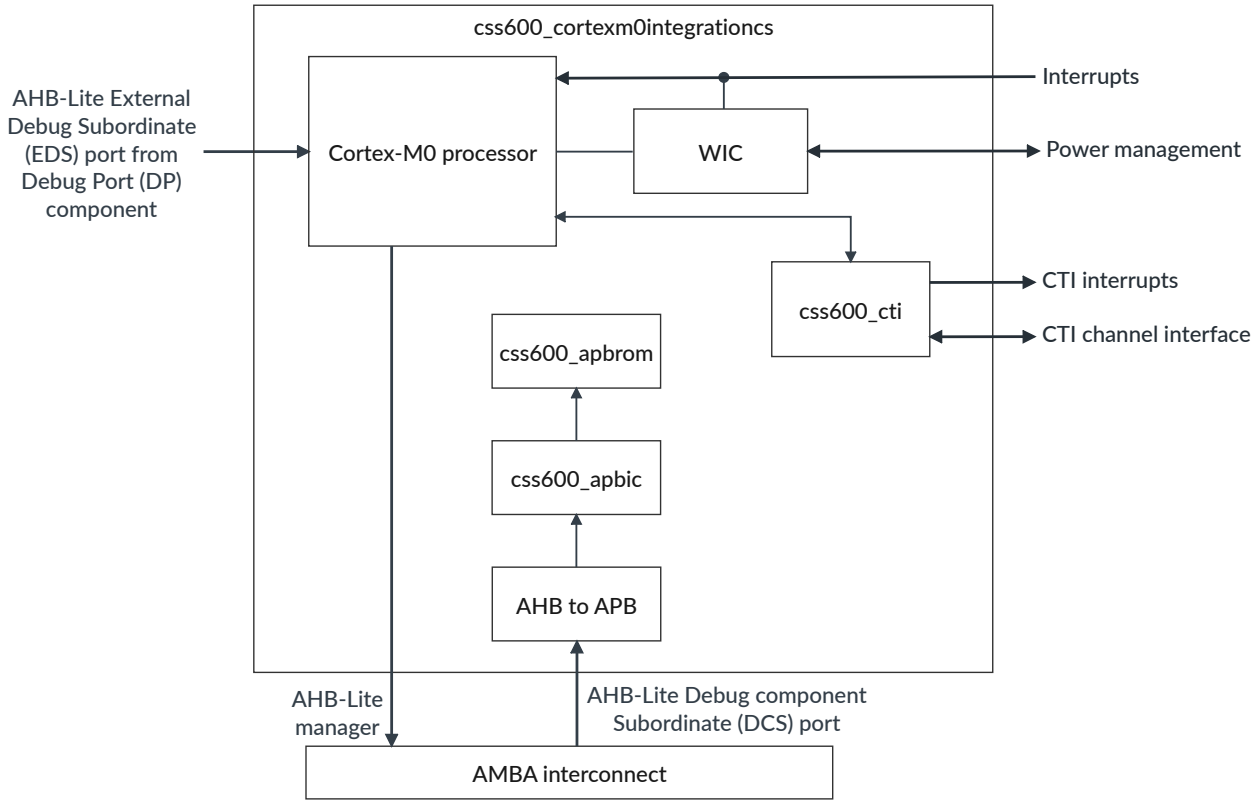
The Cortex-M0 PIL has the following interfaces:

- An AHB-Lite manager interface that connects to the system Network Interconnect (NIC)
- An AHB subordinate interface that connects to the AHB-AP port of the CoreSight™ DAP
- An AHB subordinate interface for accessing the CTI and ROM table

- Processor-specific signals such as interrupt signals, system control signals, and status signals

The following figure shows a block diagram of the Cortex-M0 PIL.

Figure 8-6: Cortex-M0 PIL block diagram



8.6.1 Cortex-M0 PIL CoreSight component identification

CoreSight™ components have several IDs that identify the components.

The following table shows the CoreSight ID register reset values for the components present within the Cortex®-M0 PIL.

Table 8-11: Cortex-M0 PIL CoreSight ID register reset values

PID	CID	DevType	DevArch	Revision	Component
0x00000004001BB4C2	0xB105900D	0x00	0x47700AF7	r0p0	css600_cortexm0integrationcs ROM table
0x00000004000BB008	0xB105E00D	0x00	0x00000000	r0p0	Arm®v6M System Control Space (SCS)
0x00000004000BB00A	0xB105E00D	0x00	0x00000000	r0p0	Armv6M Data Watchpoint and Trace (DWT)
0x00000004000BB00B	0xB105E00D	0x00	0x00000000	r0p0	Armv6M Flash Patch and Breakpoint (FPB)
0x00000004004BB9ED	0xB105900D	0x14	0x47701A14	r1p0	css600_cti

See the *Arm® CoreSight™ Architecture Specification v3.0* for information on the CoreSight ID scheme.

8.6.2 Cortex-M0 PIL Debug memory map

The debug components in the Cortex®-M0 PIL share memory space with the processor system.

You must build your system level interconnect so that the PIL Debug Component Subordinate (DCS) AHB-Lite port is accessed for the address ranges of the PIL components.

Table 8-12: Cortex-M0 PIL debug memory map

Address range	Components
0xF0000000-0xF0000FFF	PIL primary ROM table
0xF0001000-0xF0001FFF	CTI

8.7 Cortex-M3 PIL overview

The Cortex®-M3 Processor Integration Layer (PIL) integrates the processor with trace and debug components and includes suitable interfaces.

The Cortex-M3 PIL consists of the following:

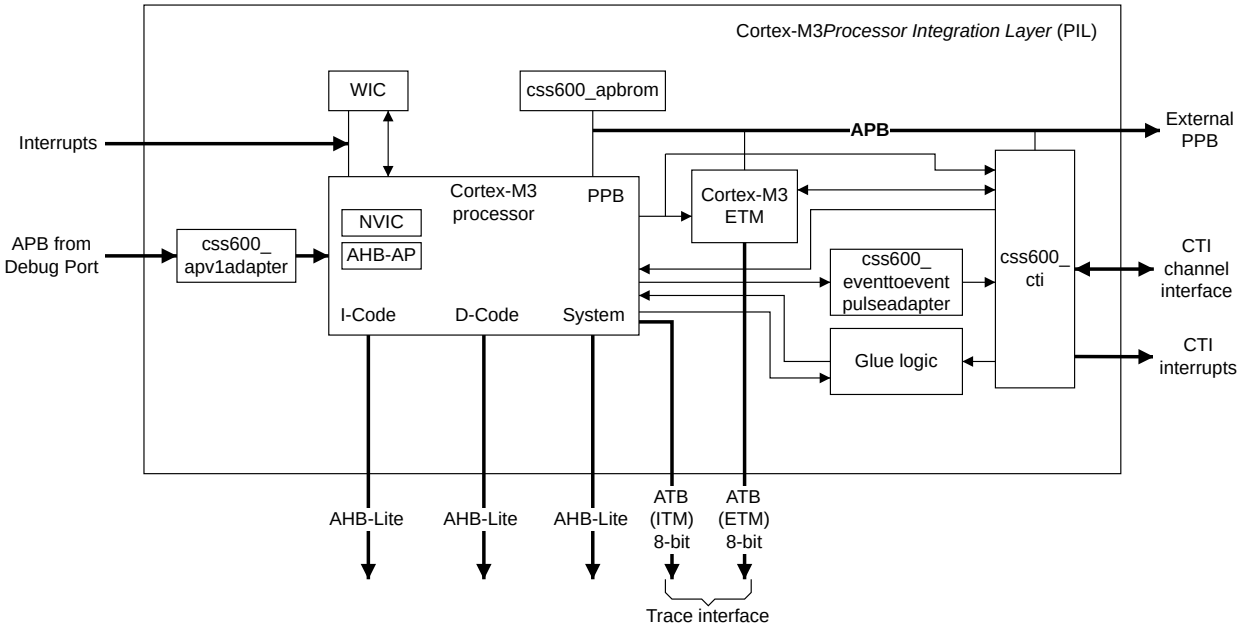
- A processor that has an Instrumentation Trace Macrocell (ITM) and AHB-(AP)
- An optional Wakeup Interrupt Controller (WIC)
- A ROM table that connects to the processor through a Private Peripheral Bus (PPB)
- An ETM trace unit that connects to the processor
- A CTI for debug event communication

The Cortex-M3 PIL supports the following external interfaces:

- AHB-Lite interfaces:
  - I-Code
  - D-Code
  - System
- Two ATB interfaces that connect to the CoreSight™ subsystem
- An APB interface for adding debug components to the PPB
- An APB interface that connects to the debug port in the CoreSight subsystem
- Processor-specific signals such as interrupt signals, system control signals, and status signals

The following figure shows a block diagram of the Cortex-M3 PIL.

Figure 8-7: Cortex-M3 PIL block diagram



8.7.1 Cortex-M3 PIL CoreSight component identification

CoreSight™ components have several IDs that identify the components.

The following table shows the CoreSight ID register reset values for the components present within the Cortex®-M3 PIL.

Table 8-13: Cortex-M3 PIL CoreSight ID register reset values

PID	CID	DevType	DevArch	Revision	Component
0x00000004001BB9E5	0xB105900D	0x00	0x47700A47	r1p0	css600_apv1adapter
0x00000004001BB4C5	0xB105900D	0x00	0x47700AF7	r0p0	css600_cortexm3integrationcs ROM table
0x00000004000BB000	0xB105E00D	0x00	0x00000000	r0p0	Arm®v7M System Control Space (SCS)
0x00000004003BB002	0xB105E00D	0x00	0x00000000	r0p0	Armv7M Data Watchpoint and Trace (DWT)
0x00000004002BB003	0xB105E00D	0x00	0x00000000	r0p0	Armv7M Flash Patch and Breakpoint (FPB)
0x00000004003BB001	0xB105E00D	0x00	0x00000000	r0p0	Armv7M Instrumentation Trace Macrocell (ITM)
0x00000004003BB924	0xB105900D	0x13	0x00000000	r0p0	Cortex-M3 ETM
0x00000004004BB9ED	0xB105900D	0x14	0x47701A14	r1p0	css600_cti

See the *Arm® CoreSight™ Architecture Specification v3.0* for information on the CoreSight ID scheme.

8.7.2 Cortex-M3 PIL Debug memory map

The debug components in the Cortex®-M3 PIL share memory space with the processor system. Part of the system memory is allocated to the Private Peripheral Bus (PPB).

The following tables show the locations of the Cortex-M3 PIL CoreSight™ components.

Table 8-14: External PPB division

Address range	Components
0xE0041000-0xE0041FFF	ETM trace unit
0xE0042000-0xE0042FFF	CTI
0xE00FF000-0xE00FFFFF	ROM table
0xE0040000-0xE0040FFF	External PPB expansion bus. In a standard single processor Cortex-M3 system, the Cortex-M3 TPIU uses this space.
0xE0043000-0xE00FFFFF	External PPB expansion bus

Table 8-15: Internal PPB division

Address range	Section	Components
0xE0000000-0xE003FFFF	Internal PPB	These components are: • Instrumentation Trace Macrocell (ITM) • Data Watchpoint and Trace (DWT) • Flash Patch and Breakpoint (FPB) • System Control Space (SCS) including for example: ◦ Nested Vectored Interrupt Controller (NVIC) ◦ SysTick ◦ Memory Protection Unit (MPU)
0xE0040000-0xE00FFFFF	External PPB	These components are: • ROM table • Embedded Trace Macrocell (ETM) trace unit • Cross Trigger Interface (CTI)

8.8 Cortex-M4 PIL overview

The Cortex®-M4 Processor Integration Layer (PIL) integrates the processor with trace and debug components and includes suitable interfaces.

The Cortex-M4 PIL consists of the following:

- A processor that has an Instrumentation Trace Macrocell (ITM) and Advanced High-performance Bus (AHB)-Access Port (AP)
- An optional Wakeup Interrupt Controller (WIC)
- A ROM table that connects to the processor through a Private Peripheral Bus (PPB)
- An Embedded Trace Macrocell (ETM) trace unit that connects to the processor

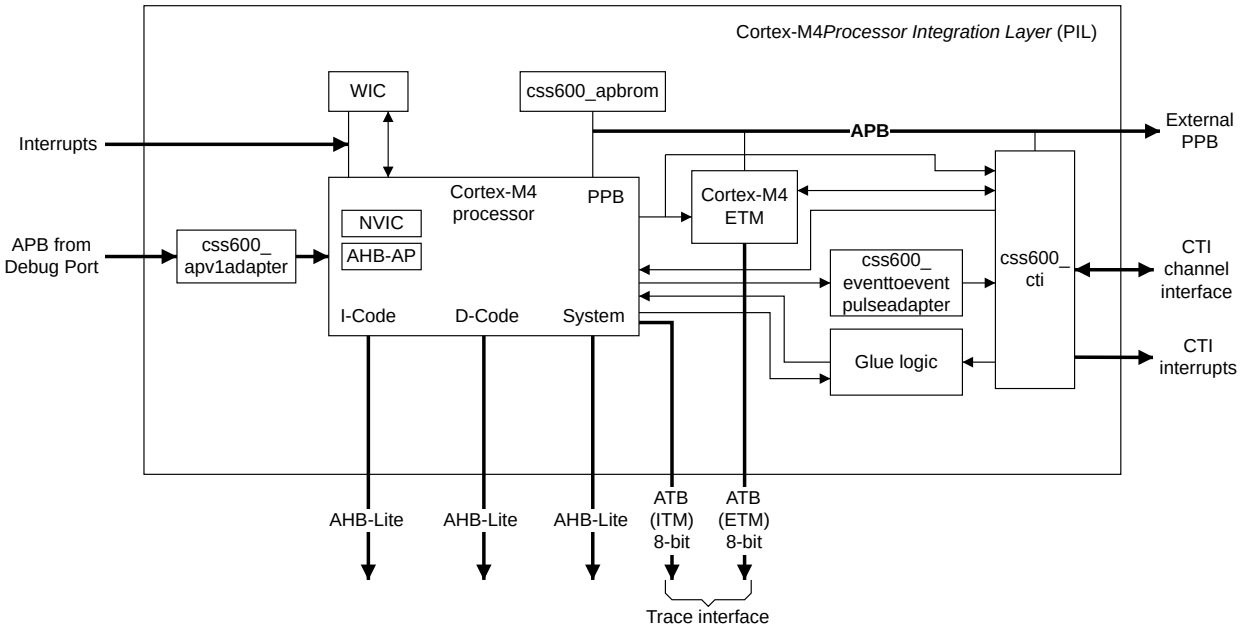
- A CTI for debug event communication

The Cortex-M4 PIL supports the following external interfaces:

- AHB-Lite interfaces:
  - I-Code
  - D-Code
  - System
- Two Advanced Trace Bus (ATB) interfaces that connect to the CoreSight™ subsystem
- An Advanced Peripheral Bus (APB) interface for adding debug components to the PPB
- An APB interface that connects to the debug port in the CoreSight subsystem
- Processor-specific signals such as interrupt signals, system control signals, and status signals

The following figure shows a block diagram of the Cortex-M4 PIL.

Figure 8-8: Cortex-M4 PIL block diagram



8.8.1 Cortex-M4 PIL CoreSight component identification

CoreSight™ components have several IDs that identify the components.

The following table shows the CoreSight ID register reset values for the components present within the Cortex®-M4 PIL.

Table 8-16: Cortex-M4 PIL CoreSight ID register reset values

PID	CID	DevType	DevArch	Revision	Component
0x00000004001BB9E5	0xB105900D	0x00	0x47700A47	r1p0	css600_apv1adapter
0x00000004001BB4C6	0xB105900D	0x00	0x47700AF7	r0p0	css600_cortexm4integrationcs ROM table
0x00000004000BB00C	0xB105E00D	0x00	0x00000000	r0p0	Arm®v7M System Control Space (SCS)
0x00000004003BB002	0xB105E00D	0x00	0x00000000	r0p0	Armv7M Data Watchpoint and Trace (DWT)
0x00000004002BB003	0xB105E00D	0x00	0x00000000	r0p0	Armv7M Flash Patch and Breakpoint (FPB)
0x00000004003BB001	0xB105E00D	0x00	0x00000000	r0p0	Armv7M Instrumentation Trace Macrocell (ITM)
0x00000004000BB925	0xB105900D	0x13	0x00000000	r0p0	Cortex-M4 ETM
0x00000004004BB9ED	0xB105900D	0x14	0x47701A14	r1p0	css600_cti

See the *Arm® CoreSight™ Architecture Specification v3.0* for information on the CoreSight ID scheme.

8.8.2 Cortex-M4 PIL Debug memory map

The debug components in the Cortex®-M4 PIL share memory space with the processor system. Part of the system memory is allocated to the Private Peripheral Bus (PPB).

The following tables show the locations of the Cortex-M4 PIL CoreSight™ components.

Table 8-17: External PPB division

Address range	Components
0xE0041000-0xE0041FFF	ETM trace unit
0xE0042000-0xE0042FFF	CTI
0xE00FF000-0xE00FFFFF	ROM table
0xE0040000-0xE0040FFF	External PPB expansion bus. In a standard single processor Cortex-M4 system, the Cortex-M4 TPIU uses this space.
0xE0043000-0xE00FFFFF	External PPB expansion bus

Table 8-18: Internal PPB division

Address range	Section	Components

0xE0000000-0xE003FFFF	Internal PPB	These components are:
		• Instrumentation Trace Macrocell (ITM)
		• Data Watchpoint and Trace (DWT)
		• Flash Patch and Breakpoint (FPB)
		• System Control Space (SCS) including for example:
		  ◦ Nested Vectored Interrupt Controller (NVIC)
		  ◦ SysTick
		  ◦ Memory Protection Unit (MPU)

Address range	Section	Components
0xE0040000-0xE00FFFFF	External PPB	These components are: • ROM table • Embedded Trace Macrocell (ETM) trace unit • Cross Trigger Interface (CTI)

9. Programmers model

The programmers model provides general information about the CoreSight SoC-600 register properties.

The following information applies to the SoC-600 components registers:

- The base address of any component is not fixed, and can be different for any particular system implementation. The offset of each register within a component from the component base address is fixed.
- Do not attempt to access reserved or unused address locations. Attempting to access these locations can result in unpredictable behavior.
- Unless otherwise stated in the accompanying text:
  - Do not modify undefined register bits.
  - Ignore undefined register bits on reads.
  - All register bits are reset to the reset value specified in the register summary table for the component.
- Access types are described as follows:

RW
Read and write.

RO
Read only.

WO
Write only.


Note

In the register summary tables and register descriptions, a reset value containing one or more '-' means that this register contains `UNKNOWN` or `IMPLEMENTATION-DEFINED` values. See the relevant register description for more information.

9.1 css600_ahbap register summary

This section describes the css600_ahbap_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-1: css600_ahbap_registers register summary

Offset	Name	Type	Reset	Width	Description
0x0	DAR0	RW	0x--------	32-bit	Direct Access Register 0
0x4	DAR1	RW	0x--------	32-bit	Direct Access Register 1

Offset	Name	Type	Reset	Width	Description
0x8	DAR2	RW	0x--------	32-bit	Direct Access Register 2
...	...				
0x3fc	DAR255	RW	0x--------	32-bit	Direct Access Register 255
0xd00	CSW	RW	0x43-000-2	32-bit	Control Status Word register
0xd04	TAR	RW	0x00000000	32-bit	Transfer Address Register
0xd0c	DRW	RW	0x--------	32-bit	Data Read/Write register
0xd10	BD0	RW	0x--------	32-bit	Banked Data register 0
0xd14	BD1	RW	0x--------	32-bit	Banked Data register 1
0xd18	BD2	RW	0x--------	32-bit	Banked Data register 2
0xd1c	BD3	RW	0x--------	32-bit	Banked Data register 3
0xd24	TRR	RW	0x00000000	32-bit	Transfer Response Register
0xdf4	CFG	RO	0x000101A0	32-bit	Configuration register
0xdf8	BASE	RO	0x-----00-	32-bit	Debug Base Address register
0xdfc	IDR	RO	0x54770008	32-bit	Identification Register
0xefc	ITSTATUS	RO	0x00000000	32-bit	Integration Test Status register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x00000003	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x000000--	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x47700A17	32-bit	Device Architecture Register
0xfcc	DEVTYPE	RO	0x00000000	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000E3	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000005B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.1.1 css600_ahbap Direct Access Register 0, DAR0

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x0`

Type

RW

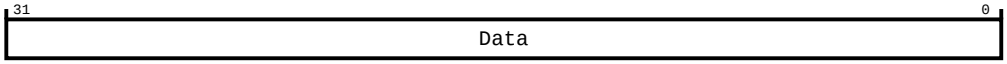
Reset value

`0x--------`

Bit descriptions

The following figure shows the DAR0 register bit assignments.

Figure 9-1: Bit assignment diagram for the DAR0 register



The following table shows the DAR0 register bit descriptions.

Table 9-2: DAR0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFC00`) + `0x0`). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.1.2 css600_ahbap Direct Access Register 1, DAR1

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x4

Type

RW

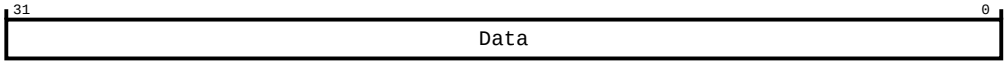
Reset value

0x--------

Bit descriptions

The following figure shows the DAR1 register bit assignments.

Figure 9-2: Bit assignment diagram for the DAR1 register



The following table shows the DAR1 register bit descriptions.

Table 9-3: DAR1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFC00) + 0x4). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.1.3 css600_ahbap Direct Access Register 2, DAR2

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x8

Type

RW

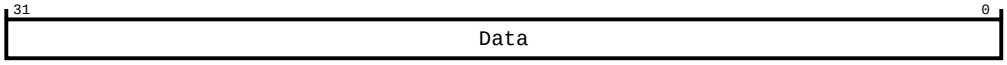
Reset value

`0x--------`

Bit descriptions

The following figure shows the DAR2 register bit assignments.

Figure 9-3: Bit assignment diagram for the DAR2 register



The following table shows the DAR2 register bit descriptions.

Table 9-4: DAR2 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFC00`) + `0x8`). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.1.4 css600_ahbap Direct Access Register 255, DAR255

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x3FC`

Type

RW

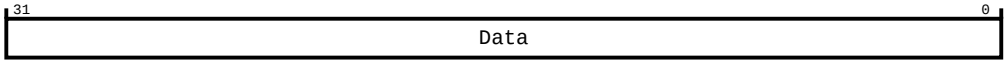
Reset value

`0x--------`

Bit descriptions

The following figure shows the DAR255 register bit assignments.

Figure 9-4: Bit assignment diagram for the DAR255 register



The following table shows the DAR255 register bit descriptions.

Table 9-5: DAR255 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFC00) + 0x3FC). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.1.5 css600_ahbap Control Status Word register, CSW

The CSW register configures and controls accesses through the Mem-AP to the connected memory system.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD00

Type

RW

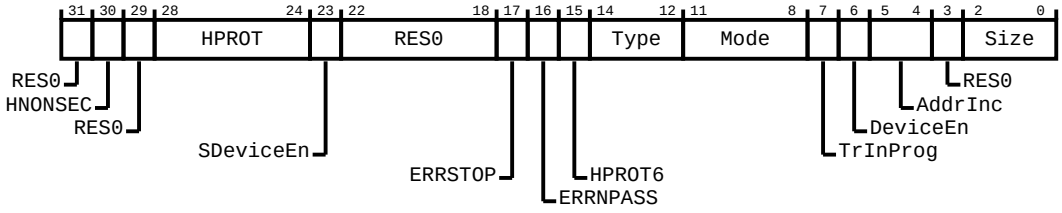
Reset value

0x43-000-2

Bit descriptions

The following figure shows the CSW register bit assignments.

Figure 9-5: Bit assignment diagram for the CSW register



The following table shows the CSW register bit descriptions.

Table 9-6: CSW bit descriptions

Bits	Name	Reset	Type	Description
31	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
30	HNONSEC	0b1	RW	Drives hnonsec_m output pin. Together with CSW.HPROT and CSW.HPROT6, CSW.HNONSEC determines the PAS of the access to be initiated by the Access Port: Non-Secure, Secure. An access will only be initiated if permitted by CSW.DeviceEn and CSW.SDeviceEn.
29	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
28:24	HPROT	0b00011	RW	Together with CSW.HPROT6, HPROT sets the protection control value to be output on hprot_m[6:0]. CSW.HPROT6 controls hprot_m[6]. CSW.HPROT controls hprot_m[4:0]. hprot_m[5] is always driven LOW. This field is reset to 0x3. The reset values of the two fields correspond to a protection value of: Non-Shareable, (Non-Allocate), Non-Lookup, Non-Modifiable, Non-Bufferable, Privileged, Data. css600_ahbap supports the following legal hprot_m encodings: **CSW.HPROT6=0b0, CSW.HPROT[4:2]=0b000** Device-nE **CSW.HPROT6=0b0, CSW.HPROT[4:2]=0b001** Device-E **CSW.HPROT6=0b0, CSW.HPROT[4:2]=0b010** Normal Non-cacheable, Non-shareable **CSW.HPROT6=0b0, CSW.HPROT[4:2]=0b110** Write-through, Non-shareable **CSW.HPROT6=0b0, CSW.HPROT[4:2]=0b111** Write-back, Non-shareable **CSW.HPROT6=0b1, CSW.HPROT[4:2]=0b010** Normal Non-cacheable, Shareable **CSW.HPROT6=0b1, CSW.HPROT[4:2]=0b110** Write-through, Shareable **CSW.HPROT6=0b1, CSW.HPROT[4:2]=0b111** Write-back, Shareable
23	SDeviceEn	**UNKNOWN**	RO	Secure Debug Enabled. This field has one of the following values: **0b0** Secure access is disabled. **0b1** Secure access is enabled.
22:18	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
17	ERRSTOP	0b0	RW	Stop on error. **0b0** Memory access errors do not prevent future memory accesses. **0b1** Memory access errors prevent future memory accesses.
16	ERRNPASS	0b0	RW	Errors are not passed upstream. **0b0** Memory access errors are passed upstream. **0b1** Memory access errors are not passed upstream.
15	HPROT6	0b0	RW	Together with CSW.HPROT, HPROT6 controls the protection value to be output on hprot_m[6:0]. This field is reset to 0x0.
14:12	Type	0b000	RO	This field is reserved. Reads return 0x0 and writes are ignored.
11:8	Mode	0b0000	RO	Mode of operation of the Mem-AP. **0b0000** Basic (normal download or upload) mode.
7	TrInProg	0b0	RO	Transfer in progress. After an ABORT operation, debug software can read this bit to check whether the aborted transaction completed. **0b0** The connection to the memory system is idle. **0b1** A transfer is in progress on the connection to the memory system.
6	DeviceEn	**UNKNOWN**	RO	Device enabled. **0b0** The Mem-AP is not enabled. **0b1** Transactions can be issued through the Mem-AP.
5:4	AddrInc	0b00	RW	Auto address increment mode on RW data access. Only increments if the current transaction completes without an error response and the transaction is not aborted. **0b00** Address auto-increment disabled. **0b01** Address increment-single enabled. **0b10** Reserved. **0b11** Reserved.
3	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
2:0	Size	0b010	RW	Size of the data access to perform. **0b000** 8 bits **0b001** 16 bits **0b010** 32 bits **0b011** Reserved **0b100** Reserved **0b101** Reserved **0b110** Reserved **0b111** Reserved

9.1.6 css600_ahbap Transfer Address Register, TAR

TAR holds the transfer address of the current transfer. TAR must be programmed before initiating any memory transfer through DRW, or Banked Data Registers, or Direct Access Registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD04

Type

RW

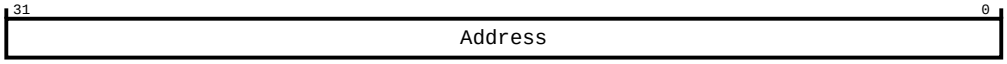
Reset value

0x00000000

Bit descriptions

The following figure shows the TAR register bit assignments.

Figure 9-6: Bit assignment diagram for the TAR register



The following table shows the TAR register bit descriptions.

Table 9-7: TAR bit descriptions

Bits	Name	Reset	Type	Description
31:0	Address	0x0	RW	Address of the current transfer. When a memory access is initiated by accessing the DRW register, the TAR value directly gives the 32-bit transfer address. When a memory access is initiated by accessing Banked Data registers, the TAR only provides the upper bits [31:4] and the remaining address bits [3:0] come from the offset of Banked Data register being accessed. When a memory access is initiated by accessing Direct Access Registers, the TAR provides the upper bits [31:10] and the remaining address bits [9:0] come from the offset of the DAR being accessed.

9.1.7 css600_ahbap Data Read/Write register, DRW

A write to the DRW register initiates a memory write transaction. AP drives DRW write data on the data bus during the data phase of the current transfer. Reading the DRW register initiates a memory read transaction.

The resulting read data that is received from the memory system is returned on the completer interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD0C

Type

RW

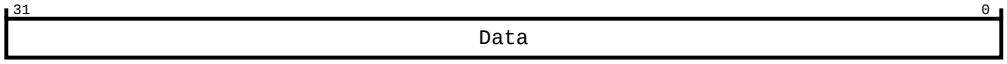
Reset value

0x--------

Bit descriptions

The following figure shows the DRW register bit assignments.

Figure 9-7: Bit assignment diagram for the DRW register



The following table shows the DRW register bit descriptions.

Table 9-8: DRW bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Current transfer data value. Writing to this register initiates a write to the address specified by the TAR. Reading this register initiates a read from the address specified by the TAR. When the read completes, the data value is returned in this register.

9.1.8 css600_ahbap Banked Data register 0, BD0

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD10`

Type

RW

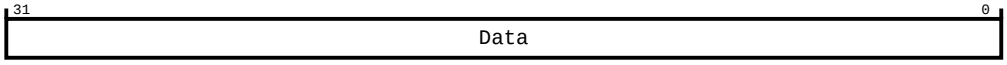
Reset value

`0x--------`

Bit descriptions

The following figure shows the BD0 register bit assignments.

Figure 9-8: Bit assignment diagram for the BD0 register



The following table shows the BD0 register bit descriptions.

Table 9-9: BD0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFFF0) + 0x0). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.1.9 css600_ahbap Banked Data register 1, BD1

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD14

Type

RW

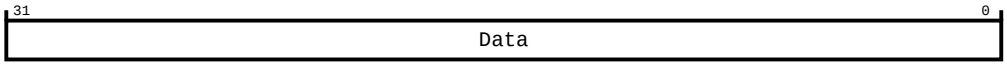
Reset value

0x--------

Bit descriptions

The following figure shows the BD1 register bit assignments.

Figure 9-9: Bit assignment diagram for the BD1 register



The following table shows the BD1 register bit descriptions.

Table 9-10: BD1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFFF0) + 0x4). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.1.10 css600_ahbap Banked Data register 2, BD2

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD18`

Type

RW

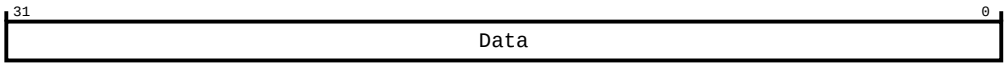
Reset value

`0x--------`

Bit descriptions

The following figure shows the BD2 register bit assignments.

Figure 9-10: Bit assignment diagram for the BD2 register



The following table shows the BD2 register bit descriptions.

Table 9-11: BD2 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFFF0`) + `0x8`). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.1.11 css600_ahbap Banked Data register 3, BD3

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD1C

Type

RW

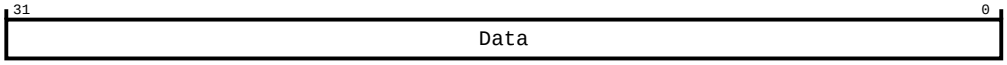
Reset value

0x--------

Bit descriptions

The following figure shows the BD3 register bit assignments.

Figure 9-11: Bit assignment diagram for the BD3 register



The following table shows the BD3 register bit descriptions.

Table 9-12: BD3 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFFF0) + 0xC). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.1.12 css600_ahbap Transfer Response Register, TRR

The Transfer Response Register is used to capture an error response received during a transaction. It is also used to clear any logged responses.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD24

Type

RW

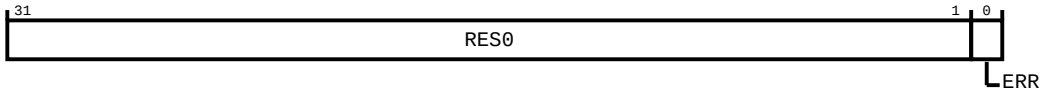
Reset value

0x00000000

Bit descriptions

The following figure shows the TRR register bit assignments.

Figure 9-12: Bit assignment diagram for the TRR register



The following table shows the TRR register bit descriptions.

Table 9-13: TRR bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	ERR	0b0	RW	Logged error. **0b0** On reads - no error response logged. Writing to this bit has no effect. **0b1** On reads - error response logged. Writing to this bit clears this bit to 0.

9.1.13 css600_ahbap Configuration register, CFG

This is the AHB-AP configuration register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDF4

Type

RO

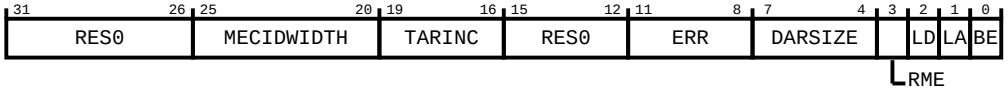
Reset value

0x000101A0

Bit descriptions

The following figure shows the CFG register bit assignments.

Figure 9-13: Bit assignment diagram for the CFG register



The following table shows the CFG register bit descriptions.

Table 9-14: CFG bit descriptions

Bits	Name	Reset	Type	Description
31:26	**RES0**	0b000000	RO	Reserved bit or field with SBZP behavior.
25:20	MECIDWIDTH	0b000000	RO	Memory Encryption Context (MEC) ID width. **0b000000** MECID not implemented.
19:16	TARINC	0b0001	RO	TAR incrementer size. **0b0001** TAR incrementer is 10-bits.
15:12	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
11:8	ERR	0b0001	RO	Identifies the type of error handling that is implemented. **0b0001** TRR, CSW.ERRNPASS and CSW.ERRSTOP are implemented.
7:4	DARSIZE	0b1010	RO	Size of DAR register space. **0b1010** DAR0-DAR255 are implemented.
3	RME	0b0	RO	Realm Management Extension **0b0** Realm Management Extension not implemented.
2	LD	0b0	RO	Large Data. Indicates support for LDE (data items greater than 32 bits). **0b0** Mem-AP does not support data items that are larger than 32 bits.
1	LA	0b0	RO	Long Address. Indicates support for LAE (greater than 32-bit of addressing). **0b0** Mem-AP only supports physical addresses of 32 bits or smaller. Registers 0xD08 and 0xDF0 are reserved.
0	BE	0b0	RO	Big-endian. Always read as 0, BE support is obsolete in Mem-AP from ADIv5.2 **0b0** Not supported.

9.1.14 css600_ahbap Debug Base Address register, BASE

Provides an initial system address for the first component in the system. Typically, the system address is the address of a top-level ROM Table that indicates where APv2 APs are located.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xDF8`

Type

RO

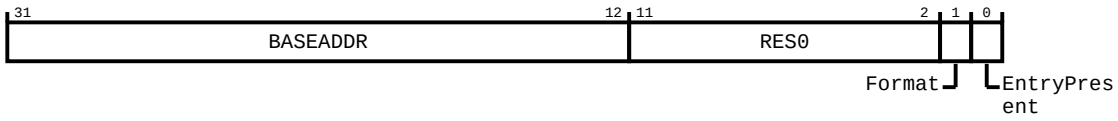
Reset value

`0x-----00-`

Bit descriptions

The following figure shows the BASE register bit assignments.

Figure 9-14: Bit assignment diagram for the BASE register



The following table shows the BASE register bit descriptions.

Table 9-15: BASE bit descriptions

Bits	Name	Reset	Type	Description
31:12	BASEADDR	**IMPLEMENTATION DEFINED**	RO	Base address of a ROM table. It points to the start of the debug register space or a ROM table address. Bits[11:0] of the address are `0x000` because the address is aligned to 4KB boundary. This field is valid only if BASE.EntryPresent bit is set to 1, in which case it returns the tie-off value of the input signal baseaddr[31:12], otherwise, it reads as `0x0`.
11:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1	Format	`0b1`	RO	Base address register format. Returns the value `0b1` indicating the ADIv5 format, which is unchanged in ADIv6.

Bits	Name	Reset	Type	Description
0	EntryPresent	IMPLEMENTATION DEFINED	RO	This field indicates whether a debug component is present for this AP. It returns the tie-off value of the input signal baseaddr_valid. **0b0** No debug entry present. **0b1** Debug entry present and BASE.BASEADDR indicate the start address of the debug register space or ROM table.

9.1.15 css600_ahbap Identification Register, IDR

This register provides a mechanism for the debugger to know various identity attributes of the AP.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xDFC`

Type

RO

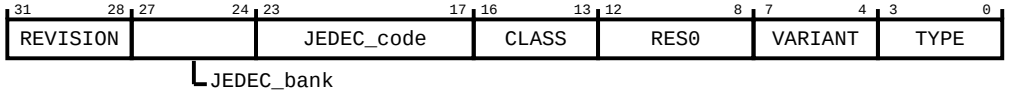
Reset value

`0x54770008`

Bit descriptions

The following figure shows the IDR register bit assignments.

Figure 9-15: Bit assignment diagram for the IDR register



The following table shows the IDR register bit descriptions.

Table 9-16: IDR bit descriptions

Bits	Name	Reset	Type	Description
31:28	REVISION	`0b0101`	RO	Revision. An incremental value starting at `0x0` for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
27:24	JEDEC_bank	`0b0100`	RO	The JEP106 continuation code. **0b0100** Arm

Bits	Name	Reset	Type	Description
23:17	JEDEC_code	0x3B	RO	The JEP106 identification code. **0x3B** Arm
16:13	CLASS	0b1000	RO	Defines the class of the AP. **0b1000** MEM-AP
12:8	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
7:4	VARIANT	0b0000	RO	Together with the TYPE field, this field identifies the AP implementation. VARIANT differentiates AP implementations that have the same value of TYPE.
3:0	TYPE	0b1000	RO	Indicates the type of bus, or other connection, that connects to the AP. **0b1000** AMBA AHB5 with enhanced HPROT.

9.1.16 css600_ahbap Integration Test Status register, ITSTATUS

This register indicates the Integration Test DP Abort status.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEFC

Type

RO

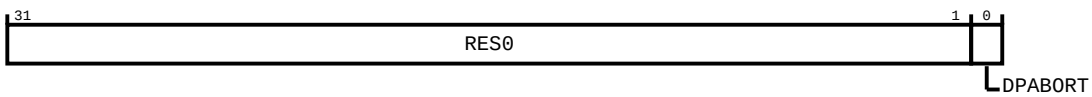
Reset value

0x00000000

Bit descriptions

The following figure shows the ITSTATUS register bit assignments.

Figure 9-16: Bit assignment diagram for the ITSTATUS register



The following table shows the ITSTATUS register bit descriptions.

Table 9-17: ITSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	DPABORT	0b0	RO	When in Integration testing mode (ITCTRL.IME=0b1): Behaves as a sticky bit and latches to 1 on a rising edge of dp_abort. Cleared on a read from this register. If dp_abort rises in the same cycle as a read of the ITSTATUS register is received, the read takes priority and the register is cleared. When in normal functional operation mode (ITCTRL.IME=0b0): Read as 0, writes ignored.

9.1.17 css600_ahbap Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

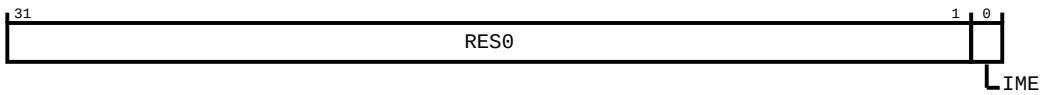
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-17: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-18: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.1.18 css600_ahbap Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

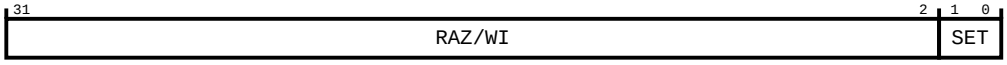
Reset value

0x00000003

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-18: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-19: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:2	RAZ/WI	0x0	RO	RAZ/WI.
1:0	SET	0b11	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.1.19 css600_ahbap Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFA4`

Type

RW

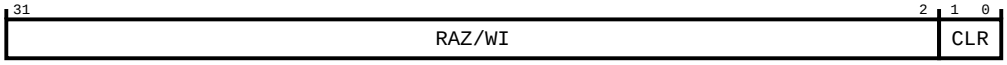
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-19: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-20: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RAZ/WI**	`0x0`	RO	**RAZ/WI.**
1:0	CLR	`0b00`	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.1.20 css600_ahbap Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

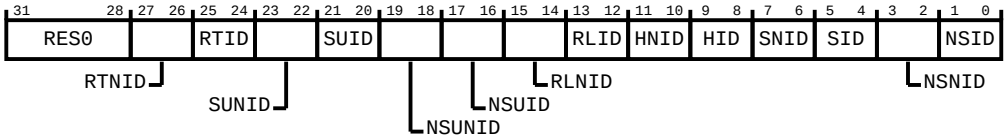
Reset value

0x000000--

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-20: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-21: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	0b00	RO	Root non-invasive debug. **0b00** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.
25:24	RTID	0b00	RO	Root invasive debug. **0b00** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	**UNKNOWN**	RO	Secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
5:4	SID	**UNKNOWN**	RO	Secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
3:2	NSNID	**UNKNOWN**	RO	Non-secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
1:0	NSID	**UNKNOWN**	RO	Non-secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.

9.1.21 css600_ahbap Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFBC`

Type

RO

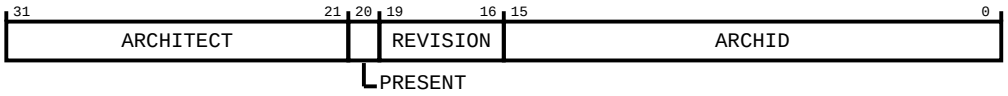
Reset value

`0x47700A17`

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-21: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-22: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	`0x23B`	RO	Defines the architect of the component **`0x23B`** Arm
20	PRESENT	`0b1`	RO	Indicates the presence of this register **`0b1`** DEVARCH is present
19:16	REVISION	`0b0000`	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	`0x0A17`	RO	Architecture ID. Returns a value that identifies the architecture of the component. **`0x0A17`** Memory Access Port v2 architecture

9.1.22 css600_ahbap Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFCC`

Type

RO

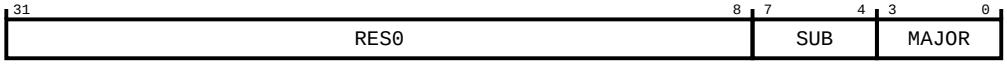
Reset value

`0x00000000`

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-22: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-23: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	`0b0000`	RO	Minor classification. Returns `0x0`, Other/undefined.
3:0	MAJOR	`0b0000`	RO	Major classification. Returns `0x0`, Miscellaneous.

9.1.23 css600_ahbap Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

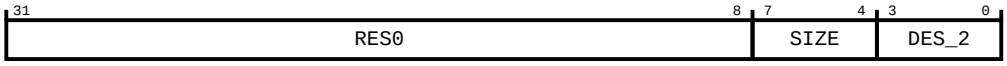
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-23: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-24: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.1.24 css600_ahbap Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

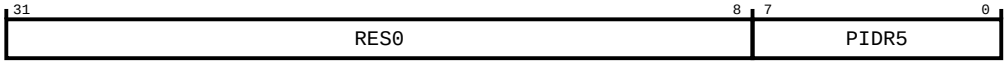
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-24: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-25: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.1.25 css600_ahbap Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD8

Type

RO

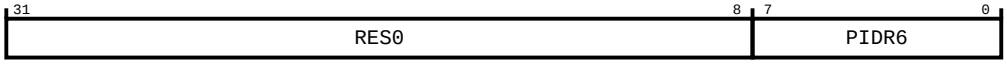
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-25: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-26: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.1.26 css600_ahbap Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

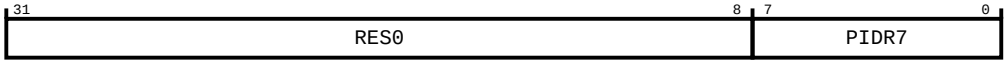
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-26: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-27: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.1.27 css600_ahbap Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

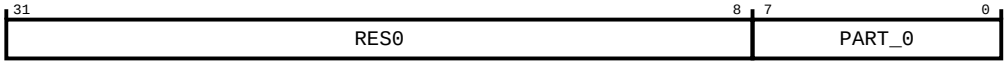
Reset value

0x000000E3

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-27: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-28: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xE3	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.1.28 css600_ahbap Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

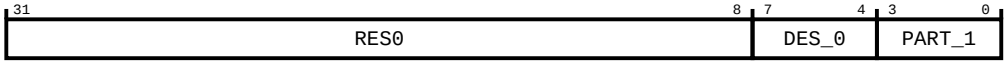
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-28: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-29: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.1.29 css600_ahbap Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

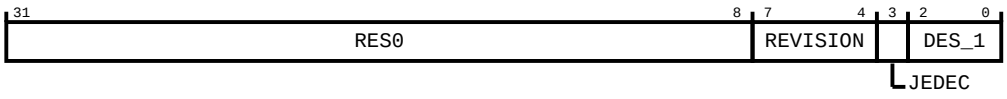
Reset value

0x0000005B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-29: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-30: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b0101	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.1.30 css600_ahbap Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFEC

Type

RO

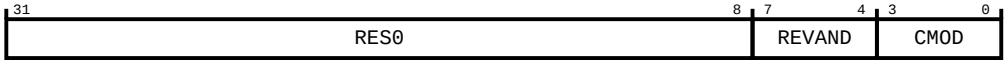
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-30: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-31: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.1.31 css600_ahbap Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

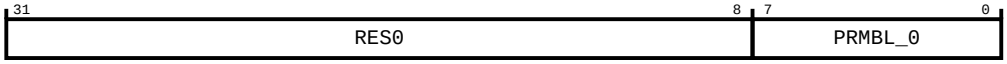
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-31: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-32: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.1.32 css600_ahbap Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF4`

Type

RO

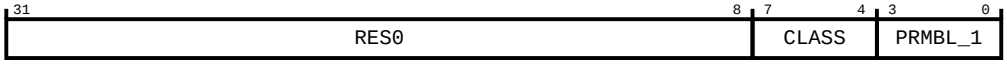
Reset value

`0x00000090`

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-32: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-33: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	`0b1001`	RO	Component class **`0b1001`** CoreSight component
3:0	PRMBL_1	`0b0000`	RO	Preamble. Returns `0x0`.

9.1.33 css600_ahbap Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

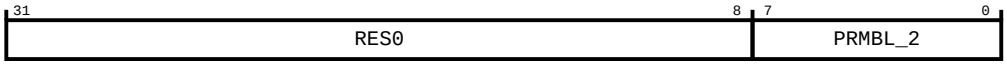
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-33: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-34: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.1.34 css600_ahbap Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

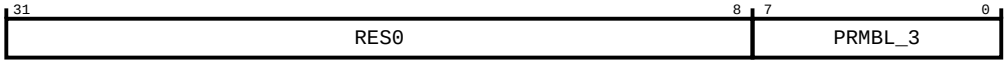
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-34: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-35: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.2 css600_apbap register summary

This section describes the css600_apbap_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-36: css600_apbap_registers register summary

Offset	Name	Type	Reset	Width	Description
0x0	DAR0	RW	0x--------	32-bit	Direct Access Register 0
0x4	DAR1	RW	0x--------	32-bit	Direct Access Register 1
0x8	DAR2	RW	0x--------	32-bit	Direct Access Register 2
...	...				
0x3fc	DAR255	RW	0x--------	32-bit	Direct Access Register 255
0xd00	CSW	RW	0x30-000-2	32-bit	Control Status Word register
0xd04	TAR	RW	0x00000000	32-bit	Transfer Address Register
0xd0c	DRW	RW	0x--------	32-bit	Data Read/Write register
0xd10	BD0	RW	0x--------	32-bit	Banked Data register 0
0xd14	BD1	RW	0x--------	32-bit	Banked Data register 1
0xd18	BD2	RW	0x--------	32-bit	Banked Data register 2
0xd1c	BD3	RW	0x--------	32-bit	Banked Data register 3
0xd24	TRR	RW	0x00000000	32-bit	Transfer Response Register
0xdf4	CFG	RO	0x000101A-	32-bit	Configuration register
0xdf8	BASE	RO	0x-----00-	32-bit	Debug Base Address register
0xdfc	IDR	RO	0x44770006	32-bit	Identification Register
0xefc	ITSTATUS	RO	0x00000000	32-bit	Integration Test Status register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x00000003	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x0-00-0--	32-bit	Authentication Status Register

Offset	Name	Type	Reset	Width	Description
0xfbc	DEVARCH	RO	0x47700A17	32-bit	Device Architecture Register
0xfcc	DEVTYPE	RO	0x00000000	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000E2	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000004B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.2.1 css600_apbap Direct Access Register 0, DAR0

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x0

Type

RW

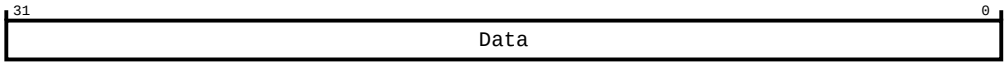
Reset value

0x--------

Bit descriptions

The following figure shows the DAR0 register bit assignments.

Figure 9-35: Bit assignment diagram for the DAR0 register



The following table shows the DAR0 register bit descriptions.

Table 9-37: DAR0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFC00) + 0x0). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.2.2 css600_apbap Direct Access Register 1, DAR1

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x4

Type

RW

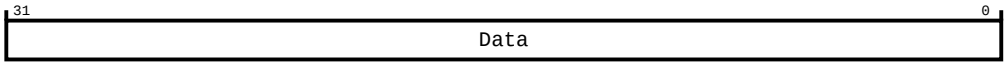
Reset value

0x--------

Bit descriptions

The following figure shows the DAR1 register bit assignments.

Figure 9-36: Bit assignment diagram for the DAR1 register



The following table shows the DAR1 register bit descriptions.

Table 9-38: DAR1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFC00) + 0x4). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.2.3 css600_apbap Direct Access Register 2, DAR2

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x8`

Type

RW

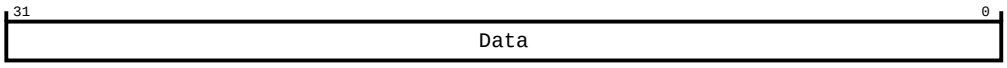
Reset value

`0x--------`

Bit descriptions

The following figure shows the DAR2 register bit assignments.

Figure 9-37: Bit assignment diagram for the DAR2 register



The following table shows the DAR2 register bit descriptions.

Table 9-39: DAR2 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFC00`) + `0x8`). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.2.4 css600_apbap Direct Access Register 255, DAR255

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x3FC`

Type

RW

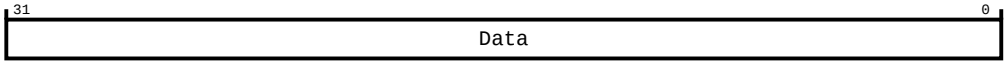
Reset value

`0x--------`

Bit descriptions

The following figure shows the DAR255 register bit assignments.

Figure 9-38: Bit assignment diagram for the DAR255 register



The following table shows the DAR255 register bit descriptions.

Table 9-40: DAR255 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFC00`) + `0x3FC`). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.2.5 css600_apbap Control Status Word register, CSW

The CSW register configures and controls accesses through the Mem-AP to the connected memory system.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD00`

Type

RW

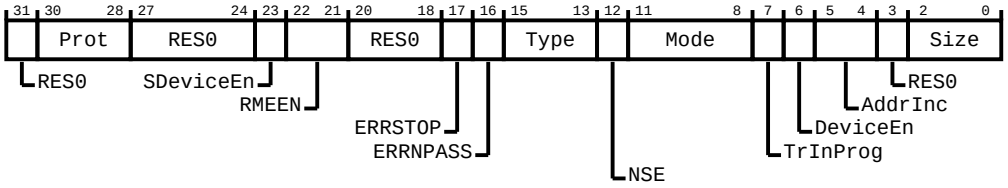
Reset value

`0x30-000-2`

Bit descriptions

The following figure shows the CSW register bit assignments.

Figure 9-39: Bit assignment diagram for the CSW register



The following table shows the CSW register bit descriptions.

Table 9-41: CSW bit descriptions

Bits	Name	Reset	Type	Description
31	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
30:28	Prot	`0b011`	RW	Drives APB requester interface pprot_r[2:0] which specifies the APB5 protection encoding. The reset value is `0x3` (Data, Non-secure, Privileged). Together with CSW.NSE, CSW.Prot[1] determines the PAS of the access to be initiated by the Access Port: Non-Secure, Secure, Realm or Root. An access will only be initiated if permitted by CSW.DeviceEn, CSW.SDeviceEn and CSW.RMEEN.
27:24	**RES0**	`0b0000`	RO	Reserved bit or field with SBZP behavior.
23	SDeviceEn	**UNKNOWN**	RO	Secure Debug Enabled. This field has one of the following values: **`0b0`** Secure access is disabled. **`0b1`** Secure access is enabled.
22:21	RMEEN	**UNKNOWN**	RO	Realm and Root access status. When legacy_tz_en==1 this field reads `0b00`. **`0b00`** Realm and Root access is disabled. **`0b01`** Realm access is enabled, Root access is disabled. **`0b10`** Reserved. **`0b11`** Realm and Root access is enabled.
20:18	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
17	ERRSTOP	0b0	RW	Stop on error. **0b0** Memory access errors do not prevent future memory accesses. **0b1** Memory access errors prevent future memory accesses.
16	ERRNPASS	0b0	RW	Errors are not passed upstream. **0b0** Memory access errors are passed upstream. **0b1** Memory access errors are not passed upstream.
15:13	Type	0b000	RO	This field is reserved. Reads return 0x0 and writes are ignored.
12	NSE	0b0	RW	Select Root and Realm. When legacy_tz_en==1 this field reads 0b0. **0b0** CSW.Prot[1] selects Secure or Non-Secure PAS. **0b1** CSW.Prot[1] selects Root or Realm PAS.
11:8	Mode	0b0000	RO	Mode of operation of the Mem-AP. **0b0000** Basic (normal download or upload) mode.
7	TrInProg	0b0	RO	Transfer in progress. After an ABORT operation, debug software can read this bit to check whether the aborted transaction completed. **0b0** The connection to the memory system is idle. **0b1** A transfer is in progress on the connection to the memory system.
6	DeviceEn	**UNKNOWN**	RO	Device enabled. **0b0** The Mem-AP is not enabled. **0b1** Transactions can be issued through the Mem-AP.

Bits	Name	Reset	Type	Description
5:4	AddrInc	0b00	RW	Auto address increment mode on RW data access. Only increments if the current transaction completes without an error response and the transaction is not aborted. **0b00** Address auto-increment disabled. **0b01** Address increment-single enabled. **0b10** Reserved. **0b11** Reserved.
3	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
2:0	Size	0b010	RO	Size of the data access to perform. **0b010** 32 bits.

9.2.6 css600_apbap Transfer Address Register, TAR

TAR holds the transfer address of the current transfer. TAR must be programmed before initiating any memory transfer through DRW, or Banked Data Registers, or Direct Access Registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD04

Type

RW

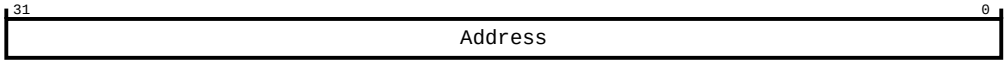
Reset value

0x00000000

Bit descriptions

The following figure shows the TAR register bit assignments.

Figure 9-40: Bit assignment diagram for the TAR register



The following table shows the TAR register bit descriptions.

Table 9-42: TAR bit descriptions

Bits	Name	Reset	Type	Description
31:0	Address	0x0	RW	Address of the current transfer. When a memory access is initiated by accessing the DRW register, the TAR value directly gives the 32-bit transfer address. When a memory access is initiated by accessing Banked Data registers, the TAR only provides the upper bits [31:4] and the remaining address bits [3:0] come from the offset of Banked Data register being accessed. When a memory access is initiated by accessing Direct Access Registers, the TAR provides the upper bits [31:10] and the remaining address bits [9:0] come from the offset of the DAR being accessed.

9.2.7 css600_apbap Data Read/Write register, DRW

A write to the DRW register initiates a memory write transaction. AP drives DRW write data on the data bus during the data phase of the current transfer. Reading the DRW register initiates a memory read transaction.

The resulting read data that is received from the memory system is returned on the completer interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD0C

Type

RW

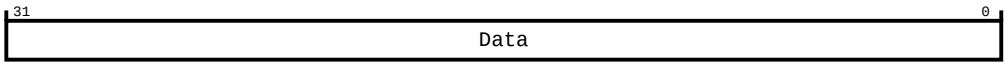
Reset value

0x--------

Bit descriptions

The following figure shows the DRW register bit assignments.

Figure 9-41: Bit assignment diagram for the DRW register



The following table shows the DRW register bit descriptions.

Table 9-43: DRW bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Current transfer data value. Writing to this register initiates a write to the address specified by the TAR. Reading this register initiates a read from the address specified by the TAR. When the read completes, the data value is returned in this register.

9.2.8 css600_apbap Banked Data register 0, BD0

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD10`

Type

RW

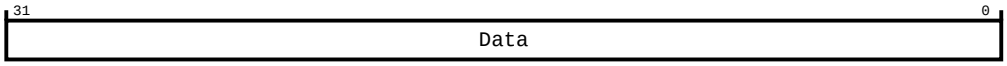
Reset value

`0x--------`

Bit descriptions

The following figure shows the BD0 register bit assignments.

Figure 9-42: Bit assignment diagram for the BD0 register



The following table shows the BD0 register bit descriptions.

Table 9-44: BD0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFFF0`) + `0x0`). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.2.9 css600_apbap Banked Data register 1, BD1

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD14`

Type

RW

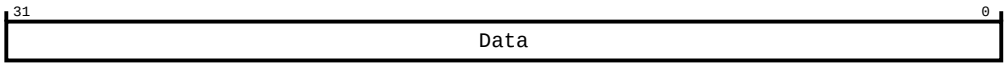
Reset value

`0x--------`

Bit descriptions

The following figure shows the BD1 register bit assignments.

Figure 9-43: Bit assignment diagram for the BD1 register



The following table shows the BD1 register bit descriptions.

Table 9-45: BD1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFFF0`) + `0x4`). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.2.10 css600_apbap Banked Data register 2, BD2

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD18

Type

RW

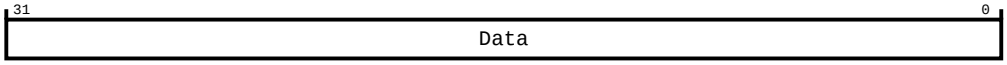
Reset value

0x--------

Bit descriptions

The following figure shows the BD2 register bit assignments.

Figure 9-44: Bit assignment diagram for the BD2 register



The following table shows the BD2 register bit descriptions.

Table 9-46: BD2 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFFF0) + 0x8). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.2.11 css600_apbap Banked Data register 3, BD3

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD1C

Type

RW

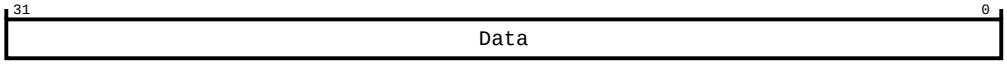
Reset value

`0x--------`

Bit descriptions

The following figure shows the BD3 register bit assignments.

Figure 9-45: Bit assignment diagram for the BD3 register



The following table shows the BD3 register bit descriptions.

Table 9-47: BD3 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFFF0`) + `0xC`). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.2.12 css600_apbap Transfer Response Register, TRR

The Transfer Response Register is used to capture an error response received during a transaction. It is also used to clear any logged responses.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD24`

Type

RW

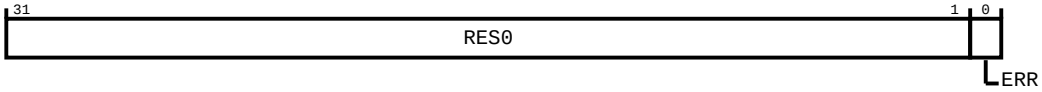
Reset value

`0x00000000`

Bit descriptions

The following figure shows the TRR register bit assignments.

Figure 9-46: Bit assignment diagram for the TRR register



The following table shows the TRR register bit descriptions.

Table 9-48: TRR bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	ERR	0b0	RW	Logged error. **0b0** On reads - no error response logged. Writing to this bit has no effect. **0b1** On reads - error response logged. Writing to this bit clears this bit to 0.

9.2.13 css600_apbap Configuration register, CFG

This is the APB-AP configuration register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDF4

Type

RO

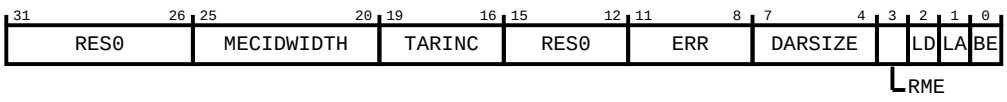
Reset value

0x000101A-

Bit descriptions

The following figure shows the CFG register bit assignments.

Figure 9-47: Bit assignment diagram for the CFG register



The following table shows the CFG register bit descriptions.

Table 9-49: CFG bit descriptions

Bits	Name	Reset	Type	Description
31:26	**RES0**	0b000000	RO	Reserved bit or field with SBZP behavior.
25:20	MECIDWIDTH	0b000000	RO	Memory Encryption Context (MEC) ID width. **0b000000** MECID not implemented.
19:16	TARINC	0b0001	RO	TAR incrementer size. **0b0001** TAR incrementer is 10-bits.
15:12	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
11:8	ERR	0b0001	RO	Identifies the type of error handling that is implemented. **0b0001** TRR, CSW.ERRNPASS and CSW.ERRSTOP are implemented.
7:4	DARSIZE	0b1010	RO	Size of DAR register space. **0b1010** DAR0-DAR255 are implemented.
3	RME	**IMPLEMENTATION DEFINED**	RO	Realm Management Extension **0b0** Realm Management Extension not implemented. **0b1** Realm Management Extension implemented.
2	LD	0b0	RO	Large Data. Indicates support for LDE (data items greater than 32 bits). **0b0** Mem-AP does not support data items that are larger than 32 bits.
1	LA	0b0	RO	Long Address. Indicates support for LAE (greater than 32-bit of addressing). **0b0** Mem-AP only supports physical addresses of 32 bits or smaller. Registers 0xD08 and 0xDF0 are reserved.
0	BE	0b0	RO	Big-endian. Always read as 0, BE support is obsolete in Mem-AP from ADIv5.2 **0b0** Not supported.

9.2.14 css600_apbap Debug Base Address register, BASE

Provides an initial system address for the first component in the system. Typically, the system address is the address of a top-level ROM Table that indicates where APv2 APs are located.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDF8

Type

RO

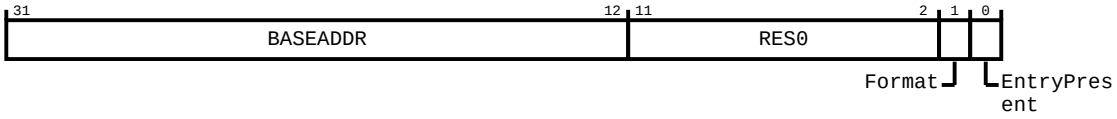
Reset value

0x-----00-

Bit descriptions

The following figure shows the BASE register bit assignments.

Figure 9-48: Bit assignment diagram for the BASE register



The following table shows the BASE register bit descriptions.

Table 9-50: BASE bit descriptions

Bits	Name	Reset	Type	Description
31:12	BASEADDR	IMPLEMENTATION DEFINED	RO	Base address of a ROM table. It points to the start of the debug register space or a ROM table address. Bits[11:0] of the address are 0x000 because the address is aligned to 4KB boundary. This field is valid only if BASE.EntryPresent bit is set to 1, in which case it returns the tie-off value of the input signal baseaddr[31:12], otherwise, it reads as 0x0.
11:2	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
1	Format	0b1	RO	Base address register format. Returns the value 0b1 indicating the ADIv5 format, which is unchanged in ADIv6.
0	EntryPresent	IMPLEMENTATION DEFINED	RO	This field indicates whether a debug component is present for this AP. It returns the tie-off value of the input signal baseaddr_valid. **0b0** No debug entry present. **0b1** Debug entry present and BASE.BASEADDR indicate the start address of the debug register space or ROM table.

9.2.15 css600_apbap Identification Register, IDR

This register provides a mechanism for the debugger to know various identity attributes of the AP.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDFC

Type

RO

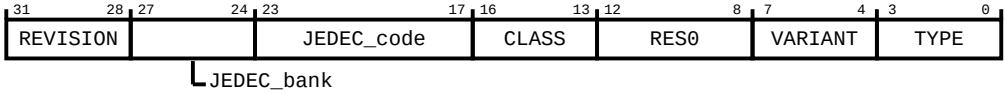
Reset value

0x44770006

Bit descriptions

The following figure shows the IDR register bit assignments.

Figure 9-49: Bit assignment diagram for the IDR register



The following table shows the IDR register bit descriptions.

Table 9-51: IDR bit descriptions

Bits	Name	Reset	Type	Description
31:28	REVISION	0b0100	RO	Revision. An incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
27:24	JEDEC_bank	0b0100	RO	The JEP106 continuation code. **0b0100** Arm
23:17	JEDEC_code	0x3B	RO	The JEP106 identification code. **0x3B** Arm
16:13	CLASS	0b1000	RO	Defines the class of the AP. **0b1000** MEM-AP
12:8	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
7:4	VARIANT	0b0000	RO	Together with the TYPE field, this field identifies the AP implementation. VARIANT differentiates AP implementations that have the same value of TYPE.
3:0	TYPE	0b0110	RO	Indicates the type of bus, or other connection, that connects to the AP. **0b0110** AMBA APB4 or APB5 bus.

9.2.16 css600_apbap Integration Test Status register, ITSTATUS

This register indicates the Integration Test DP Abort status.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEFC`

Type

RO

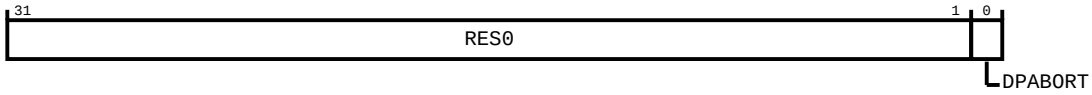
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITSTATUS register bit assignments.

Figure 9-50: Bit assignment diagram for the ITSTATUS register



The following table shows the ITSTATUS register bit descriptions.

Table 9-52: ITSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
0	DPABORT	`0b0`	RO	When in Integration testing mode (ITCTRL.IME=`0b1`): Behaves as a sticky bit and latches to 1 on a rising edge of dp_abort. Cleared on a read from this register. If dp_abort rises in the same cycle as a read of the ITSTATUS register is received, the read takes priority and the register is cleared. When in normal functional operation mode (ITCTRL.IME=`0b0`): Read as 0, writes ignored.

9.2.17 css600_apbap Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xF00`

Type

RW

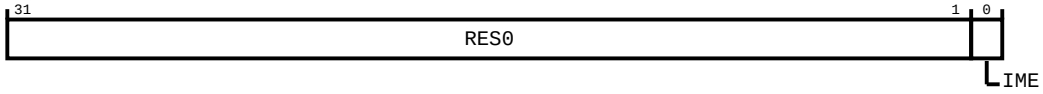
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-51: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-53: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
0	IME	`0b0`	RW	Integration Mode Enable. **`0b0`** The component must enter functional mode. **`0b1`** The component must enter integration mode, and enable support for topology detection and integration testing.

9.2.18 css600_apbap Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

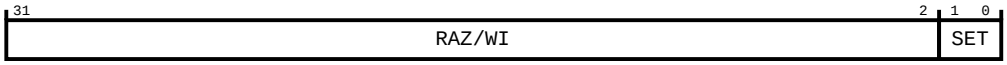
Reset value

0x00000003

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-52: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-54: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:2	RAZ/WI	0x0	RO	RAZ/WI.
1:0	SET	0b11	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.2.19 css600_apbap Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA4

Type

RW

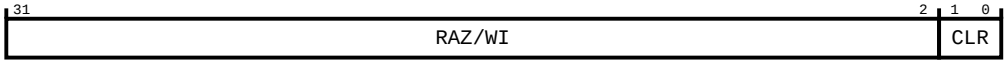
Reset value

0x00000000

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-53: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-55: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RAZ/WI**	`0x0`	RO	**RAZ/WI**.
1:0	CLR	`0b00`	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.2.20 css600_apbap Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFB8`

Type

RO

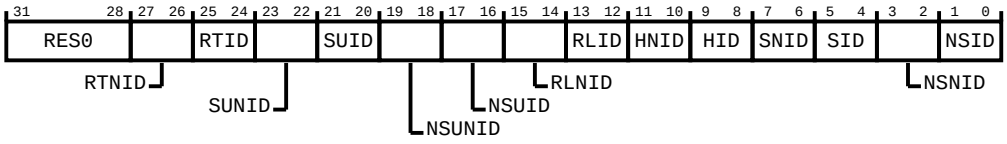
Reset value

`0x0-00-0--`

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-54: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-56: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	**UNKNOWN**	RO	Root non-invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
25:24	RTID	**UNKNOWN**	RO	Root invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	**UNKNOWN**	RO	Realm non-invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
13:12	RLID	**UNKNOWN**	RO	Realm invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
7:6	SNID	**UNKNOWN**	RO	Secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
5:4	SID	**UNKNOWN**	RO	Secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
3:2	NSNID	**UNKNOWN**	RO	Non-secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
1:0	NSID	**UNKNOWN**	RO	Non-secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.

9.2.21 css600_apbap Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFBC`

Type

RO

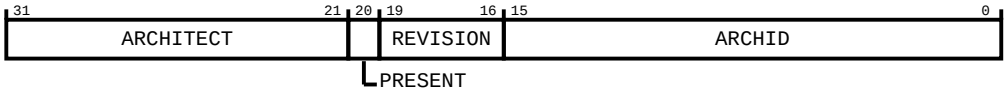
Reset value

`0x47700A17`

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-55: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-57: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x23B	RO	Defines the architect of the component **0x23B** Arm
20	PRESENT	0b1	RO	Indicates the presence of this register **0b1** DEVARCH is present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0A17	RO	Architecture ID. Returns a value that identifies the architecture of the component. **0x0A17** Memory Access Port v2 architecture

9.2.22 css600_apbap Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

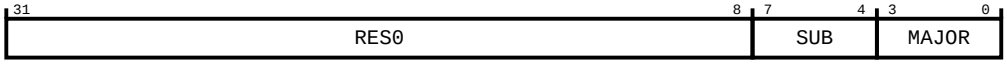
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-56: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-58: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	0b0000	RO	Minor classification. Returns 0x0, Other/undefined.
3:0	MAJOR	0b0000	RO	Major classification. Returns 0x0, Miscellaneous.

9.2.23 css600_apbap Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

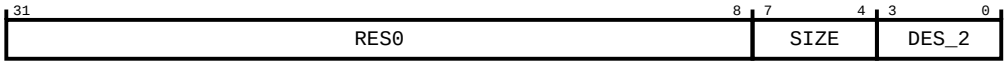
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-57: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-59: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.2.24 css600_apbap Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

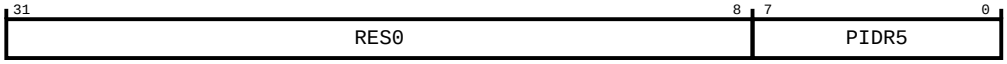
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-58: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-60: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.2.25 css600_apbap Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

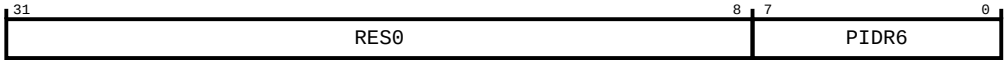
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-59: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-61: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.2.26 css600_apbap Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFDC`

Type

RO

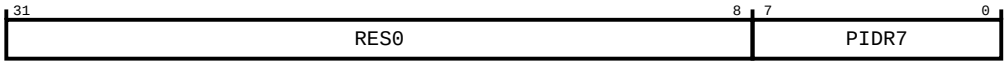
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-60: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-62: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.2.27 css600_apbap Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

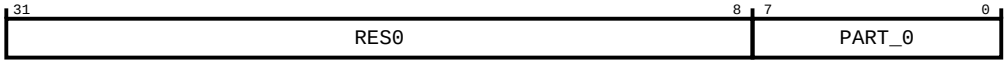
Reset value

0x000000E2

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-61: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-63: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xE2	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.2.28 css600_apbap Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

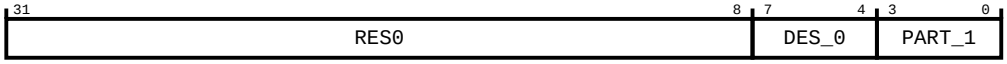
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-62: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-64: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.2.29 css600_apbap Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

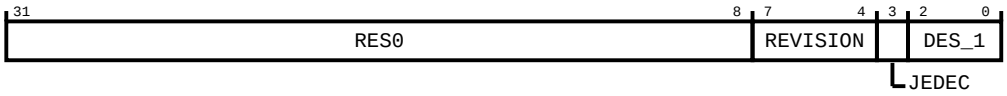
Reset value

0x0000004B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-63: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-65: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b0100	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.2.30 css600_apbap Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

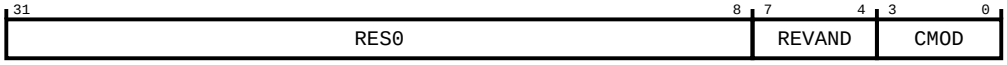
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-64: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-66: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	`0b0000`	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is `0x0`.
3:0	CMOD	`0b0000`	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is `0x0`.

9.2.31 css600_apbap Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

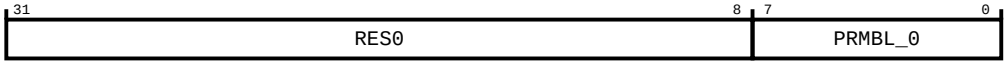
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-65: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-67: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.2.32 css600_apbap Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

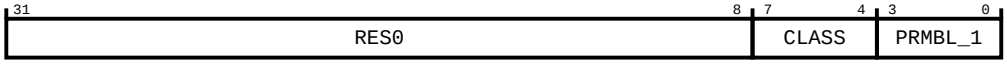
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-66: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-68: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.2.33 css600_apbap Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

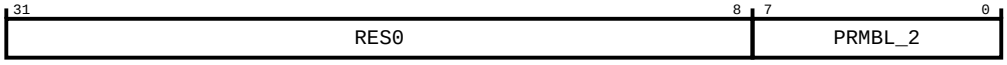
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-67: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-69: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.2.34 css600_apbap Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

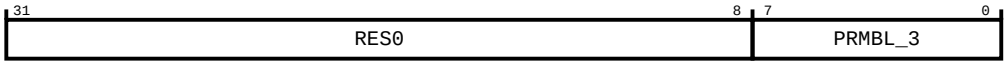
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-68: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-70: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.3 css600_apbrom introduction

The SoC-600 css600_apbrom component can be configured to be 32-bit or 64-bit.

The selected configuration affects the number of ROMEntry registers, as follows:

- The 32-bit configuration has 512 32-bit ROMEntry registers.
- The 64-bit configuration has 256 ROMEntry registers, each split into a lower and a higher 32-bit register.

9.4 css600_apbrom_32bit register summary

This section describes the css600_apbrom_32bit_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-71: css600_apbrom_32bit_registers register summary

Offset	Name	Type	Reset	Width	Description
0x0	ROMEntry0	RO	0x--------	32-bit	ROM Entries register 0
0x4	ROMEntry1	RO	0x--------	32-bit	ROM Entries register 1
0x8	ROMEntry2	RO	0x--------	32-bit	ROM Entries register 2
...	...				
0x7fc	ROMEntry511	RO	0x--------	32-bit	ROM Entries register 511
0xfbc	DEVARCH	RO	0x47700AF7	32-bit	Device Architecture Register
0xfc8	DEVID	RO	0x0000000-	32-bit	Device Configuration Register
0xfd0	PIDR4	RO	0x0000000-	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000--	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000--	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x000000--	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.4.1 css600_apbrom_32bit ROM Entries register 0, ROMEntry0

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x0

Type

RO

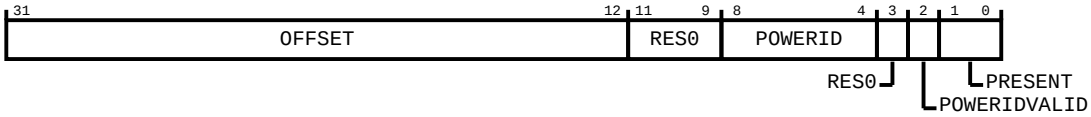
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry0 register bit assignments.

Figure 9-69: Bit assignment diagram for the ROMEntry0 register



The following table shows the ROMEntry0 register bit descriptions.

Table 9-72: ROMEntry0 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	0b000	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **0b0** POWERID field of this register is not valid **0b1** POWERID field of this register is valid

Bits	Name	Reset	Type	Description
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **0b00** ROM table entry not present. This is the last entry. **0b01** Reserved **0b10** ROM table entry not present. This is not the last entry. **0b11** ROM table entry present

9.4.2 css600_apbrom_32bit ROM Entries register 1, ROMEntry1

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x4`

Type

RO

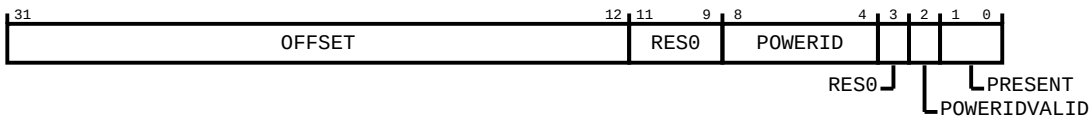
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry1 register bit assignments.

Figure 9-70: Bit assignment diagram for the ROMEntry1 register



The following table shows the ROMEntry1 register bit descriptions.

Table 9-73: ROMEntry1 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **0b0** POWERID field of this register is not valid **0b1** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **0b00** ROM table entry not present. This is the last entry. **0b01** Reserved **0b10** ROM table entry not present. This is not the last entry. **0b11** ROM table entry present

9.4.3 css600_apbrom_32bit ROM Entries register 2, ROMEntry2

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x8`

Type

RO

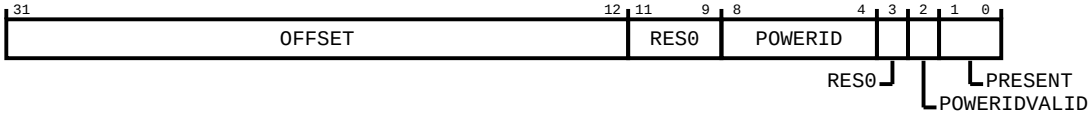
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry2 register bit assignments.

Figure 9-71: Bit assignment diagram for the ROMEntry2 register



The following table shows the ROMEntry2 register bit descriptions.

Table 9-74: ROMEntry2 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **`0b0`** POWERID field of this register is not valid **`0b1`** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **`0b00`** ROM table entry not present. This is the last entry. **`0b01`** Reserved **`0b10`** ROM table entry not present. This is not the last entry. **`0b11`** ROM table entry present

9.4.4 css600_apbrom_32bit ROM Entries register 511, ROMEntry511

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x7FC`

Type

RO

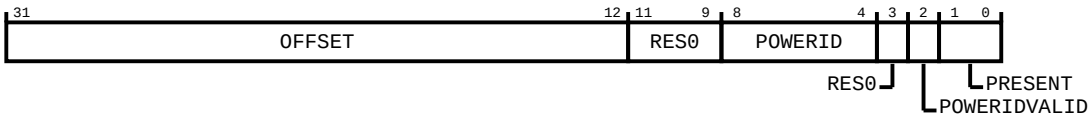
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry511 register bit assignments.

Figure 9-72: Bit assignment diagram for the ROMEntry511 register



The following table shows the ROMEntry511 register bit descriptions.

Table 9-75: ROMEntry511 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **0b0** POWERID field of this register is not valid **0b1** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **0b00** ROM table entry not present. This is the last entry. **0b01** Reserved **0b10** ROM table entry not present. This is not the last entry. **0b11** ROM table entry present

9.4.5 css600_apbrom_32bit Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFBC`

Type

RO

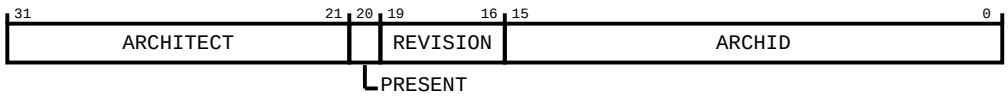
Reset value

`0x47700AF7`

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-73: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-76: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x23B	RO	Defines the architect of the component **0x23B** Arm
20	PRESENT	0b1	RO	Indicates the presence of this register **0b1** DEVARCH is present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0AF7	RO	Architecture ID. Returns a value that identifies the architecture of the component. **0x0AF7** CoreSight ROM architecture

9.4.6 css600_apbrom_32bit Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC8

Type

RO

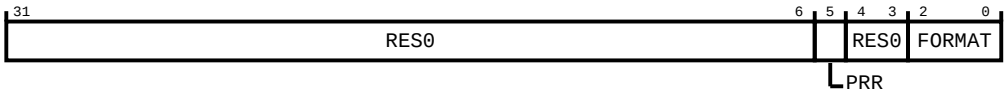
Reset value

0x0000000-

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-74: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-77: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:6	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
5	PRR	0b0	RO	Indicates that power request functionality is included. **0b0** GPR is not included
4:3	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
2:0	FORMAT	**IMPLEMENTATION DEFINED**	RO	ROM format. **0b000** 32-bit format **0b001** 64-bit format

9.4.7 css600_apbrom_32bit Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

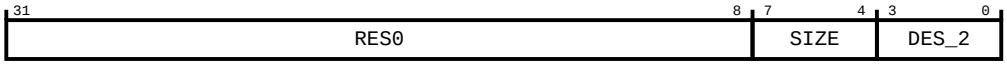
Reset value

0x0000000-

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-75: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-78: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	**IMPLEMENTATION DEFINED**	RO	JEP106 continuation code, bits [3:0]. Set by the configuration inputs jep106_if[3:0]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.4.8 css600_apbrom_32bit Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

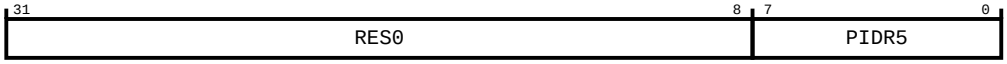
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-76: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-79: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.4.9 css600_apbrom_32bit Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

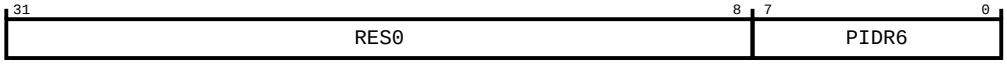
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-77: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-80: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.4.10 css600_apbrom_32bit Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFDC`

Type

RO

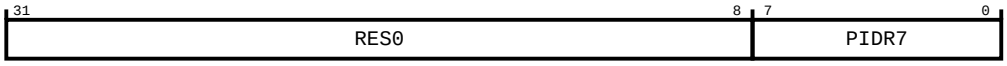
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-78: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-81: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.4.11 css600_apbrom_32bit Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

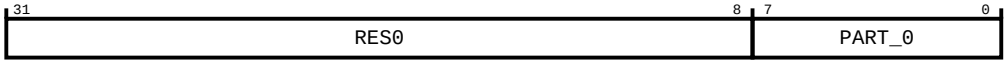
Reset value

0x000000--

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-79: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-82: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	**IMPLEMENTATION DEFINED**	RO	Part number, bits[7:0]. Set by the configuration inputs part_number[7:0]

9.4.12 css600_apbrom_32bit Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

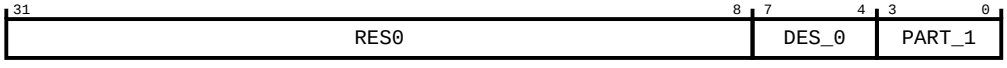
Reset value

0x000000--

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-80: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-83: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	DES_0	**IMPLEMENTATION DEFINED**	RO	JEP106 identification code, bits[3:0]. Set by the configuration inputs jep106_id[3:0]. Together, with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	**IMPLEMENTATION DEFINED**	RO	Part number, bits[11:8]. Set by the configuration inputs part_number[11:8].

9.4.13 css600_apbrom_32bit Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFE8`

Type

RO

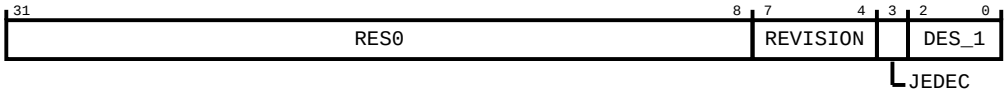
Reset value

`0x000000--`

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-81: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-84: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	**IMPLEMENTATION DEFINED**	RO	Revision. Set by the configuration inputs revision[3:0].
3	JEDEC	`0b1`	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	**IMPLEMENTATION DEFINED**	RO	JEP106 identification code, bits[6:4]. Set by the configuration inputs jep106_id[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.4.14 css600_apbrom_32bit Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

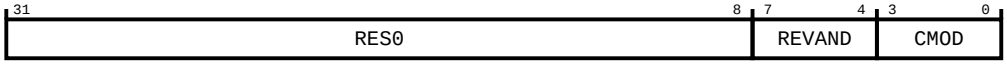
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-82: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-85: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	`0b0000`	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is `0x0`.
3:0	CMOD	`0b0000`	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is `0x0`.

9.4.15 css600_apbrom_32bit Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

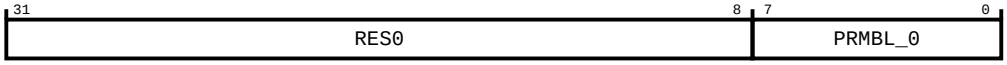
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-83: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-86: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.4.16 css600_apbrom_32bit Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

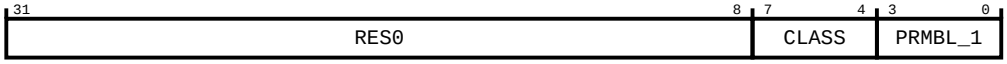
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-84: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-87: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.4.17 css600_apbrom_32bit Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

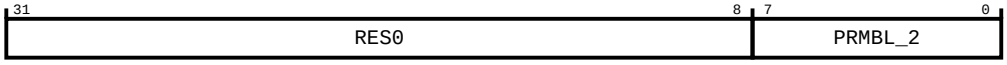
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-85: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-88: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.4.18 css600_apbrom_32bit Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

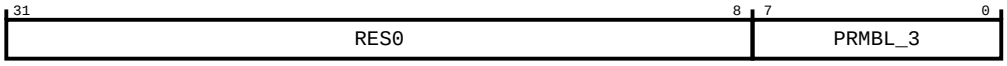
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-86: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-89: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.5 css600_apbrom_64bit register summary

This section describes the css600_apbrom_64bit_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-90: css600_apbrom_64bit_registers register summary

Offset	Name	Type	Reset	Width	Description
0x0	ROMEntry0	RO	0x--------	32-bit	ROM Entries register 0
0x4	ROMEntry0HI	RO	0x--------	32-bit	ROM Entries register 0 bits [63:32]
0x8	ROMEntry1	RO	0x--------	32-bit	ROM Entries register 1
0xc	ROMEntry1HI	RO	0x--------	32-bit	ROM Entries register 1 bits [63:32]
0x10	ROMEntry2	RO	0x--------	32-bit	ROM Entries register 2
0x14	ROMEntry2HI	RO	0x--------	32-bit	ROM Entries register 2 bits [63:32]
...	...				
0x7f8	ROMEntry255	RO	0x--------	32-bit	ROM Entries register 255
0x7fc	ROMEntry255HI	RO	0x--------	32-bit	ROM Entries register 255 bits [63:32]
0xfbc	DEVARCH	RO	0x47700AF7	32-bit	Device Architecture Register
0xfc8	DEVID	RO	0x0000000-	32-bit	Device Configuration Register
0xfd0	PIDR4	RO	0x0000000-	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000--	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000--	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x000000--	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.5.1 css600_apbrom_64bit ROM Entries register 0, ROMEntry0

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x0

Type

RO

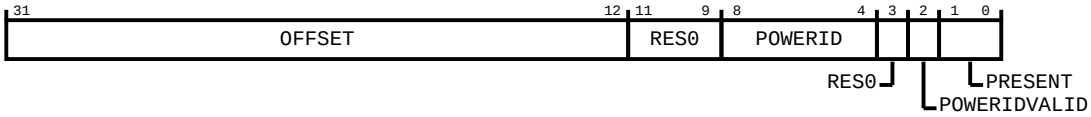
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry0 register bit assignments.

Figure 9-87: Bit assignment diagram for the ROMEntry0 register



The following table shows the ROMEntry0 register bit descriptions.

Table 9-91: ROMEntry0 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	0b000	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **0b0** POWERID field of this register is not valid **0b1** POWERID field of this register is valid

Bits	Name	Reset	Type	Description
1:0	PRESENT	IMPLEMENTATION DEFINED	RO	Indicates whether the ROM table entry is present: **0b00** ROM table entry not present. This is the last entry. **0b01** Reserved **0b10** ROM table entry not present. This is not the last entry. **0b11** ROM table entry present

9.5.2 css600_apbrom_64bit ROM Entries register 0 bits [63:32], ROMEntry0HI

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x4

Type

RO

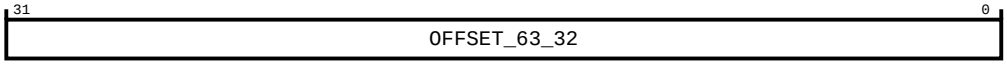
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry0HI register bit assignments.

Figure 9-88: Bit assignment diagram for the ROMEntry0HI register



The following table shows the ROMEntry0HI register bit descriptions.

Table 9-92: ROMEntry0HI bit descriptions

Bits	Name	Reset	Type	Description
31:0	OFFSET_63_32	IMPLEMENTATION DEFINED	RO	Upper 32 bits of the component address offset.

9.5.3 css600_apbrom_64bit ROM Entries register 1, ROMEntry1

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x8

Type

RO

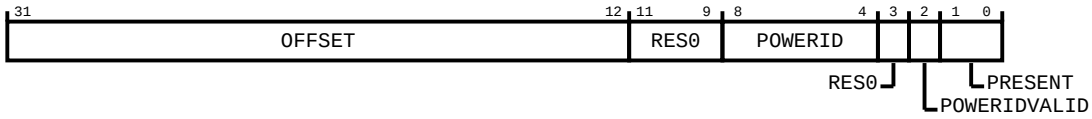
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry1 register bit assignments.

Figure 9-89: Bit assignment diagram for the ROMEntry1 register



The following table shows the ROMEntry1 register bit descriptions.

Table 9-93: ROMEntry1 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	0b000	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **0b0** POWERID field of this register is not valid **0b1** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **0b00** ROM table entry not present. This is the last entry. **0b01** Reserved **0b10** ROM table entry not present. This is not the last entry. **0b11** ROM table entry present

9.5.4 css600_apbrom_64bit ROM Entries register 1 bits [63:32], ROMEntry1HI

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC

Type

RO

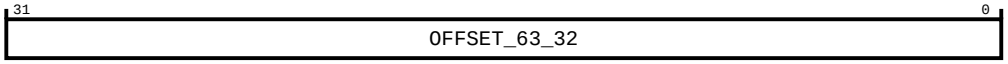
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry1HI register bit assignments.

Figure 9-90: Bit assignment diagram for the ROMEntry1HI register



The following table shows the ROMEntry1HI register bit descriptions.

Table 9-94: ROMEntry1HI bit descriptions

Bits	Name	Reset	Type	Description
31:0	OFFSET_63_32	**IMPLEMENTATION DEFINED**	RO	Upper 32 bits of the component address offset.

9.5.5 css600_apbrom_64bit ROM Entries register 2, ROMEntry2

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x10`

Type

RO

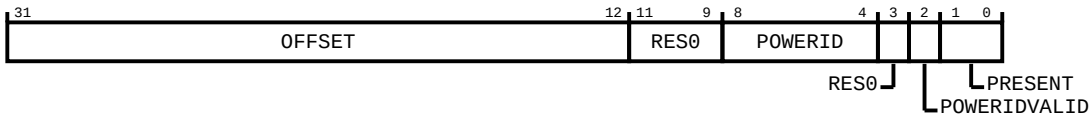
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry2 register bit assignments.

Figure 9-91: Bit assignment diagram for the ROMEntry2 register



The following table shows the ROMEntry2 register bit descriptions.

Table 9-95: ROMEntry2 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **`0b0`** POWERID field of this register is not valid **`0b1`** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **`0b00`** ROM table entry not present. This is the last entry. **`0b01`** Reserved **`0b10`** ROM table entry not present. This is not the last entry. **`0b11`** ROM table entry present

9.5.6 css600_apbrom_64bit ROM Entries register 2 bits [63:32], ROMEntry2HI

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x14`

Type

RO

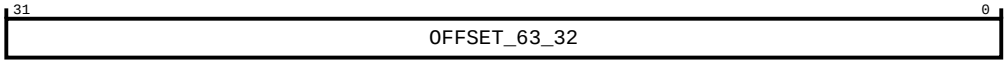
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry2HI register bit assignments.

Figure 9-92: Bit assignment diagram for the ROMEntry2HI register



The following table shows the ROMEntry2HI register bit descriptions.

Table 9-96: ROMEntry2HI bit descriptions

Bits	Name	Reset	Type	Description
31:0	OFFSET_63_32	IMPLEMENTATION DEFINED	RO	Upper 32 bits of the component address offset.

9.5.7 css600_apbrom_64bit ROM Entries register 255, ROMEntry255

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x7F8

Type

RO

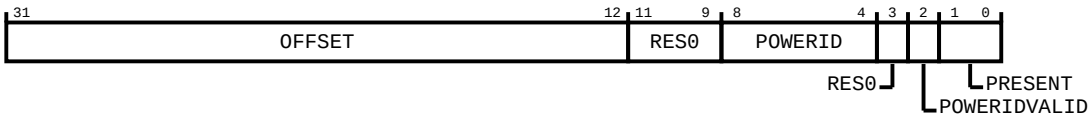
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry255 register bit assignments.

Figure 9-93: Bit assignment diagram for the ROMEntry255 register



The following table shows the ROMEntry255 register bit descriptions.

Table 9-97: ROMEntry255 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **0b0** POWERID field of this register is not valid **0b1** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **0b00** ROM table entry not present. This is the last entry. **0b01** Reserved **0b10** ROM table entry not present. This is not the last entry. **0b11** ROM table entry present

9.5.8 css600_apbrom_64bit ROM Entries register 255 bits [63:32], ROMEntry255HI

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x7FC`

Type

RO

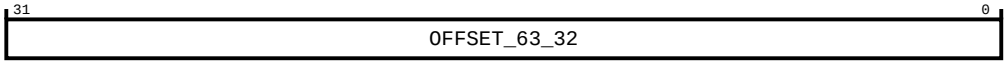
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry255HI register bit assignments.

Figure 9-94: Bit assignment diagram for the ROMEntry255HI register



The following table shows the ROMEntry255HI register bit descriptions.

Table 9-98: ROMEntry255HI bit descriptions

Bits	Name	Reset	Type	Description
31:0	OFFSET_63_32	IMPLEMENTATION DEFINED	RO	Upper 32 bits of the component address offset.

9.5.9 css600_apbrom_64bit Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFBC

Type

RO

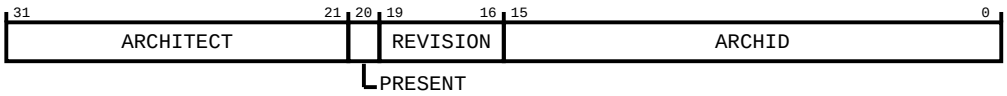
Reset value

0x47700AF7

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-95: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-99: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x23B	RO	Defines the architect of the component **0x23B** Arm
20	PRESENT	0b1	RO	Indicates the presence of this register **0b1** DEVARCH is present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0AF7	RO	Architecture ID. Returns a value that identifies the architecture of the component. **0x0AF7** CoreSight ROM architecture

9.5.10 css600_apbrom_64bit Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC8

Type

RO

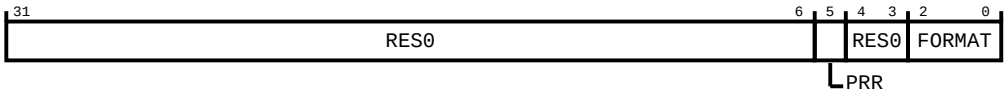
Reset value

0x0000000-

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-96: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-100: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:6	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
5	PRR	0b0	RO	Indicates that power request functionality is included. **0b0** GPR is not included
4:3	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
2:0	FORMAT	**IMPLEMENTATION DEFINED**	RO	ROM format. **0b000** 32-bit format **0b001** 64-bit format

9.5.11 css600_apbrom_64bit Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

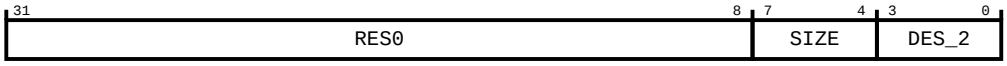
Reset value

0x0000000-

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-97: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-101: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	**IMPLEMENTATION DEFINED**	RO	JEP106 continuation code, bits [3:0]. Set by the configuration inputs jep106_if[3:0]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.5.12 css600_apbrom_64bit Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

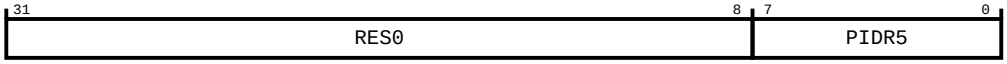
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-98: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-102: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.5.13 css600_apbrom_64bit Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

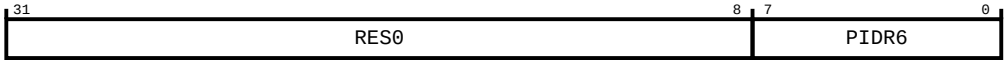
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-99: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-103: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.5.14 css600_apbrom_64bit Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFDC`

Type

RO

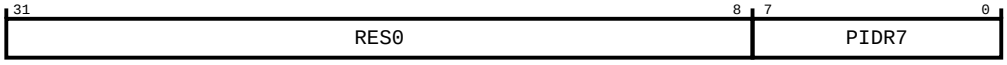
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-100: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-104: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.5.15 css600_apbrom_64bit Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

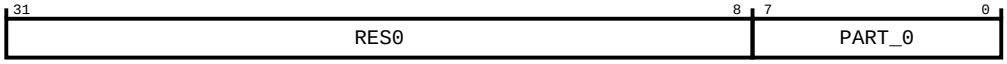
Reset value

0x000000--

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-101: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-105: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	IMPLEMENTATION DEFINED	RO	Part number, bits[7:0]. Set by the configuration inputs part_number[7:0]

9.5.16 css600_apbrom_64bit Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

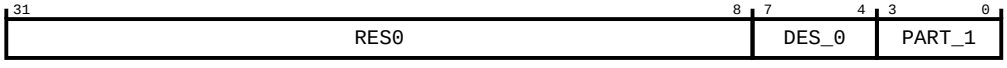
Reset value

0x000000--

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-102: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-106: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	DES_0	**IMPLEMENTATION DEFINED**	RO	JEP106 identification code, bits[3:0]. Set by the configuration inputs jep106_id[3:0]. Together, with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	**IMPLEMENTATION DEFINED**	RO	Part number, bits[11:8]. Set by the configuration inputs part_number[11:8].

9.5.17 css600_apbrom_64bit Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFE8`

Type

RO

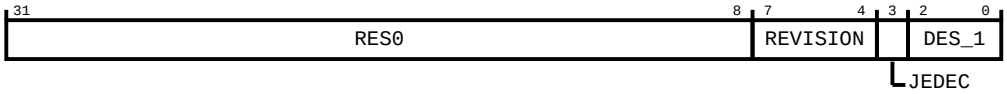
Reset value

`0x000000--`

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-103: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-107: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	**IMPLEMENTATION DEFINED**	RO	Revision. Set by the configuration inputs revision[3:0].
3	JEDEC	`0b1`	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	**IMPLEMENTATION DEFINED**	RO	JEP106 identification code, bits[6:4]. Set by the configuration inputs jep106_id[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.5.18 css600_apbrom_64bit Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

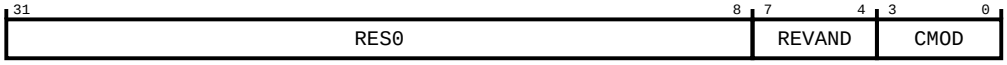
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-104: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-108: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	`0b0000`	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is `0x0`.
3:0	CMOD	`0b0000`	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is `0x0`.

9.5.19 css600_apbrom_64bit Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

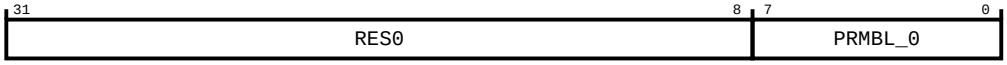
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-105: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-109: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.5.20 css600_apbrom_64bit Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

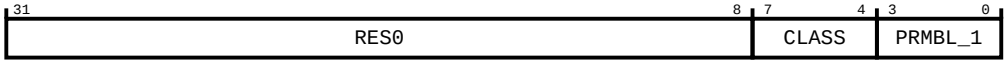
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-106: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-110: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.5.21 css600_apbrom_64bit Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

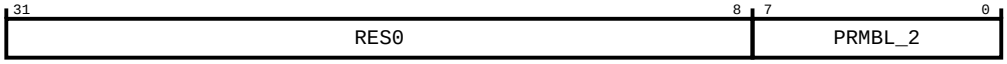
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-107: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-111: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.5.22 css600_apbrom_64bit Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

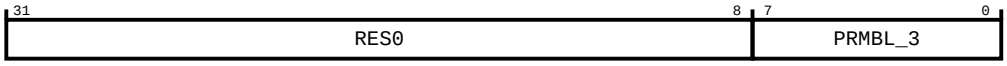
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-108: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-112: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.6 css600_apbrom_gpr introduction

The SoC-600 css600_apbrom_gpr component can be configured to be 32-bit or 64-bit.

The selected configuration affects the number of ROMEntry registers, as follows:

- The 32-bit configuration has 512 32-bit ROMEntry registers.
- The 64-bit configuration has 256 ROMEntry registers, each split into a lower and a higher 32-bit register.

9.7 css600_apbrom_gpr_32bit register summary

This section describes the css600_apbrom_gpr_32bit_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-113: css600_apbrom_gpr_32bit_registers register summary

Offset	Name	Type	Reset	Width	Description
0x0	ROMEntry0	RO	0x--------	32-bit	ROM Entries register 0
0x4	ROMEntry1	RO	0x--------	32-bit	ROM Entries register 1
0x8	ROMEntry2	RO	0x--------	32-bit	ROM Entries register 2
...	...				
0x7fc	ROMEntry511	RO	0x--------	32-bit	ROM Entries register 511
0xa00	DBGPCR0	RW	0x00000001	32-bit	Debug Power Control Register 0
0xa04	DBGPCR1	RW	0x0000000-	32-bit	Debug Power Control Register 1
0xa08	DBGPCR2	RW	0x0000000-	32-bit	Debug Power Control Register 2
...	...				
0xa7c	DBGPCR31	RW	0x0000000-	32-bit	Debug Power Control Register 31
0xa80	DBGPSR0	RO	0x0000000-	32-bit	Debug Power Status Register 0
0xa84	DBGPSR1	RO	0x0000000-	32-bit	Debug Power Status Register 1
0xa88	DBGPSR2	RO	0x0000000-	32-bit	Debug Power Status Register 2
...	...				
0xafc	DBGPSR31	RO	0x0000000-	32-bit	Debug Power Status Register 31
0xb00	SYSPCR0	RW	0x00000001	32-bit	System Power Control Register 0
0xb04	SYSPCR1	RW	0x0000000-	32-bit	System Power Control Register 1
0xb08	SYSPCR2	RW	0x0000000-	32-bit	System Power Control Register 2
...	...				
0xb7c	SYSPCR31	RW	0x0000000-	32-bit	System Power Control Register 31
0xb80	SYSPSR0	RO	0x0000000-	32-bit	System Power Status Register 0
0xb84	SYSPSR1	RO	0x0000000-	32-bit	System Power Status Register 1
0xb88	SYSPSR2	RO	0x0000000-	32-bit	System Power Status Register 2
...	...				
0xbfc	SYSPSR31	RO	0x0000000-	32-bit	System Power Status Register 31
0xc00	PRIDR0	RO	0x00000031	32-bit	Power Request ID Register
0xc10	DBGRSTRR	RW	0x00000000	32-bit	Debug Reset Request Register
0xc14	DBGRSTAR	RO	0x00000000	32-bit	Debug Reset Acknowledge Register
0xc18	SYSRSTRR	RW	0x00000000	32-bit	System Reset Request Register

Offset	Name	Type	Reset	Width	Description
0xc1c	SYSRSTAR	RO	0x00000000	32-bit	System Reset Acknowledge Register
0xfb8	AUTHSTATUS	RO	0x0-00-0--	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x47700AF7	32-bit	Device Architecture Register
0xfc8	DEVID	RO	0x0000002-	32-bit	Device Configuration Register
0xfd0	PIDR4	RO	0x0000000-	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000--	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000--	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x000000--	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.7.1 css600_apbrom_gpr_32bit ROM Entries register 0, ROMEntry0

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x0

Type

RO

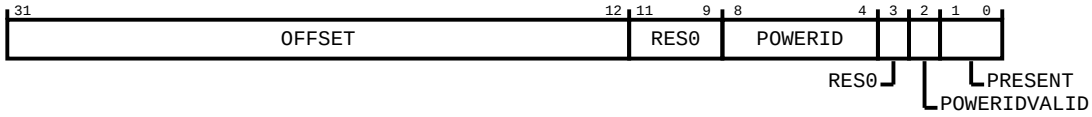
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry0 register bit assignments.

Figure 9-109: Bit assignment diagram for the ROMEntry0 register



The following table shows the ROMEntry0 register bit descriptions.

Table 9-114: ROMEntry0 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **`0b0`** POWERID field of this register is not valid **`0b1`** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **`0b00`** ROM table entry not present. This is the last entry. **`0b01`** Reserved **`0b10`** ROM table entry not present. This is not the last entry. **`0b11`** ROM table entry present

9.7.2 css600_apbrom_gpr_32bit ROM Entries register 1, ROMEntry1

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x4

Type

RO

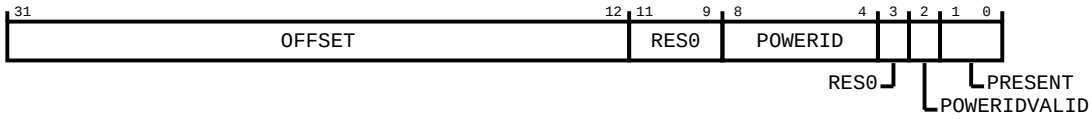
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry1 register bit assignments.

Figure 9-110: Bit assignment diagram for the ROMEntry1 register



The following table shows the ROMEntry1 register bit descriptions.

Table 9-115: ROMEntry1 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	0b000	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **0b0** POWERID field of this register is not valid **0b1** POWERID field of this register is valid

Bits	Name	Reset	Type	Description
1:0	PRESENT	IMPLEMENTATION DEFINED	RO	Indicates whether the ROM table entry is present: **0b00** ROM table entry not present. This is the last entry. **0b01** Reserved **0b10** ROM table entry not present. This is not the last entry. **0b11** ROM table entry present

9.7.3 css600_apbrom_gpr_32bit ROM Entries register 2, ROMEntry2

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x8

Type

RO

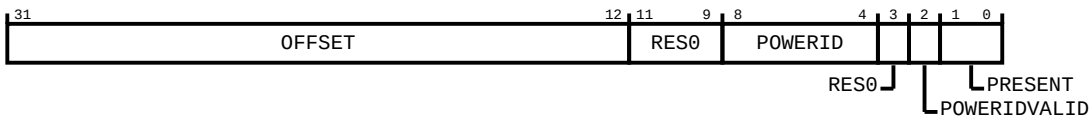
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry2 register bit assignments.

Figure 9-111: Bit assignment diagram for the ROMEntry2 register



The following table shows the ROMEntry2 register bit descriptions.

Table 9-116: ROMEntry2 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **`0b0`** POWERID field of this register is not valid **`0b1`** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **`0b00`** ROM table entry not present. This is the last entry. **`0b01`** Reserved **`0b10`** ROM table entry not present. This is not the last entry. **`0b11`** ROM table entry present

9.7.4 css600_apbrom_gpr_32bit ROM Entries register 511, ROMEntry511

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x7FC`

Type

RO

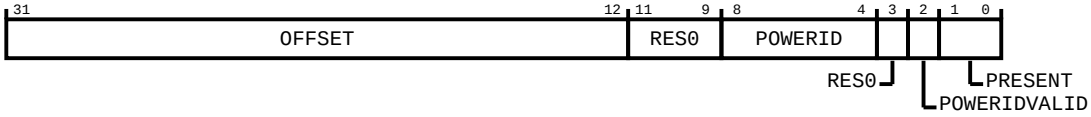
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry511 register bit assignments.

Figure 9-112: Bit assignment diagram for the ROMEntry511 register



The following table shows the ROMEntry511 register bit descriptions.

Table 9-117: ROMEntry511 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **`0b0`** POWERID field of this register is not valid **`0b1`** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **`0b00`** ROM table entry not present. This is the last entry. **`0b01`** Reserved **`0b10`** ROM table entry not present. This is not the last entry. **`0b11`** ROM table entry present

9.7.5 css600_apbrom_gpr_32bit Debug Power Control Register 0, DBGPCR0

The DBGPCR0 register indicates whether power has been requested for debug domain 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xA00`

Type

RW

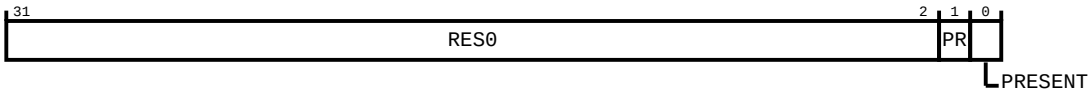
Reset value

`0x00000001`

Bit descriptions

The following figure shows the DBGPCR0 register bit assignments.

Figure 9-113: Bit assignment diagram for the DBGPCR0 register



The following table shows the DBGPCR0 register bit descriptions.

Table 9-118: DBGPCR0 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1	PR	`0b0`	RW	Power request. Software can set this bit when non-invasive debug is enabled. **`0b0`** Indicates that power is not requested for debug domain 0 **`0b1`** Indicates that power is requested for debug domain 0
0	PRESENT	`0b1`	RO	Indicates the presence of power domain control for debug domain 0: **`0b1`** Indicates that the power request is implemented for debug domain 0

9.7.6 css600_apbrom_gpr_32bit Debug Power Control Register 1, DBGPCR1

The DBGPCR1 register indicates whether power has been requested for debug domain 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xA04`

Type

RW

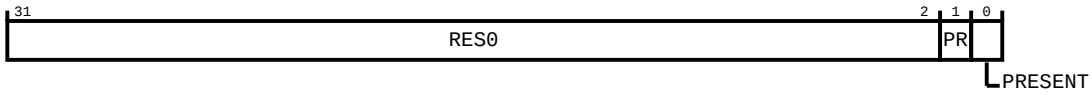
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the DBGPCR1 register bit assignments.

Figure 9-114: Bit assignment diagram for the DBGPCR1 register



The following table shows the DBGPCR1 register bit descriptions.

Table 9-119: DBGPCR1 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1	PR	`0b0`	RW	Power request. Reserved if DBGPCR1.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **`0b0`** Indicates that power is not requested for debug domain 1 **`0b1`** Indicates that power is requested for debug domain 1
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for debug domain 1: **`0b0`** Indicates that the power request is not implemented for debug domain 1 **`0b1`** Indicates that the power request is implemented for debug domain 1

9.7.7 css600_apbrom_gpr_32bit Debug Power Control Register 2, DBGPCR2

The DBGPCR2 register indicates whether power has been requested for debug domain 2.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xA08

Type

RW

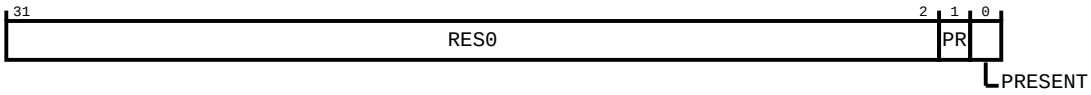
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPCR2 register bit assignments.

Figure 9-115: Bit assignment diagram for the DBGPCR2 register



The following table shows the DBGPCR2 register bit descriptions.

Table 9-120: DBGPCR2 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	PR	0b0	RW	Power request. Reserved if DBGPCR2.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **0b0** Indicates that power is not requested for debug domain 2 **0b1** Indicates that power is requested for debug domain 2
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for debug domain 2: **0b0** Indicates that the power request is not implemented for debug domain 2 **0b1** Indicates that the power request is implemented for debug domain 2

9.7.8 css600_apbrom_gpr_32bit Debug Power Control Register 31, DBGPCR31

The DBGPCR31 register indicates whether power has been requested for debug domain 31.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xA7C`

Type

RW

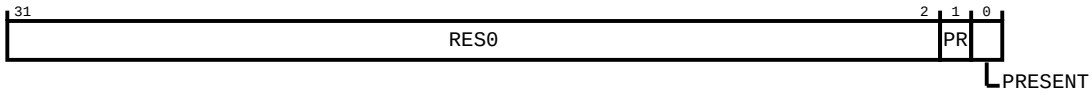
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the DBGPCR31 register bit assignments.

Figure 9-116: Bit assignment diagram for the DBGPCR31 register



The following table shows the DBGPCR31 register bit descriptions.

Table 9-121: DBGPCR31 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1	PR	`0b0`	RW	Power request. Reserved if DBGPCR31.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **`0b0`** Indicates that power is not requested for debug domain 31 **`0b1`** Indicates that power is requested for debug domain 31
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for debug domain 31: **`0b0`** Indicates that the power request is not implemented for debug domain 31 **`0b1`** Indicates that the power request is implemented for debug domain 31

9.7.9 css600_apbrom_gpr_32bit Debug Power Status Register 0, DBGPSR0

The DBGPSR0 register indicates the power status for debug domain 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xA80`

Type

RO

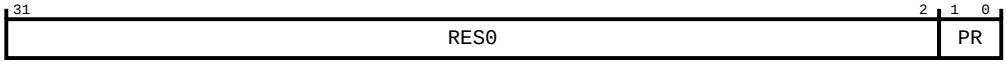
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the DBGPSR0 register bit assignments.

Figure 9-117: Bit assignment diagram for the DBGPSR0 register



The following table shows the DBGPSR0 register bit descriptions.

Table 9-122: DBGPSR0 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1:0	PR	**UNKNOWN**	RO	Power status. **`0b00`** Debug domain 0 might not be powered. **`0b11`** Debug domain 0 is powered and must remain powered until DBGPCR0.PR == 0.

9.7.10 css600_apbrom_gpr_32bit Debug Power Status Register 1, DBGPSR1

The DBGPSR1 register indicates the power status for debug domain 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xA84

Type

RO

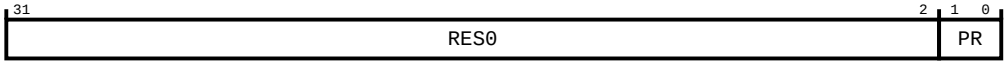
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPSR1 register bit assignments.

Figure 9-118: Bit assignment diagram for the DBGPSR1 register



The following table shows the DBGPSR1 register bit descriptions.

Table 9-123: DBGPSR1 bit descriptions

Bits	Name	Reset	Type	Description
31:2	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	UNKNOWN	RO	Power status. **0b00** Debug domain 1 might not be powered. **0b11** Debug domain 1 is powered and must remain powered until DBGPCR1.PR == 0.

9.7.11 css600_apbrom_gpr_32bit Debug Power Status Register 2, DBGPSR2

The DBGPSR2 register indicates the power status for debug domain 2.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xA88

Type

RO

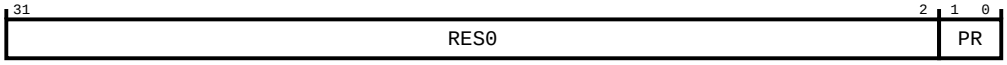
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPSR2 register bit assignments.

Figure 9-119: Bit assignment diagram for the DBGPSR2 register



The following table shows the DBGPSR2 register bit descriptions.

Table 9-124: DBGPSR2 bit descriptions

Bits	Name	Reset	Type	Description
31:2	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	UNKNOWN	RO	Power status. **0b00** Debug domain 2 might not be powered. **0b11** Debug domain 2 is powered and must remain powered until DBGPCR2.PR == 0.

9.7.12 css600_apbrom_gpr_32bit Debug Power Status Register 31, DBGPSR31

The DBGPSR31 register indicates the power status for debug domain 31.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xAFC

Type

RO

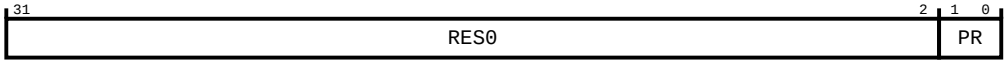
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPSR31 register bit assignments.

Figure 9-120: Bit assignment diagram for the DBGPSR31 register



The following table shows the DBGPSR31 register bit descriptions.

Table 9-125: DBGPSR31 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	**UNKNOWN**	RO	Power status. **0b00** Debug domain 31 might not be powered. **0b11** Debug domain 31 is powered and must remain powered until DBGPCR31.PR == 0.

9.7.13 css600_apbrom_gpr_32bit System Power Control Register 0, SYSPCR0

The SYSPCR0 register indicates whether power has been requested for system domain 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xB00

Type

RW

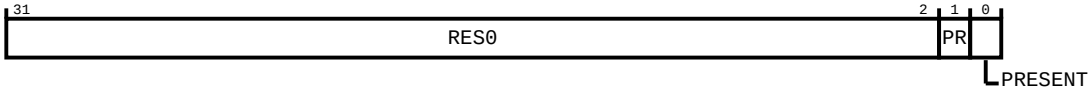
Reset value

0x00000001

Bit descriptions

The following figure shows the SYSPCR0 register bit assignments.

Figure 9-121: Bit assignment diagram for the SYSPCR0 register



The following table shows the SYSPCR0 register bit descriptions.

Table 9-126: SYSPCR0 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	PR	0b0	RW	Power request. Software can set this bit when non-invasive debug is enabled. **0b0** Indicates that power is not requested for system domain 0 **0b1** Indicates that power is requested for system domain 0
0	PRESENT	0b1	RO	Indicates the presence of power domain control for system domain 0 **0b1** Indicates that the power request is implemented for system domain 0

9.7.14 css600_apbrom_gpr_32bit System Power Control Register 1, SYSPCR1

The SYSPCR1 register indicates whether power has been requested for system domain 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xB04

Type

RW

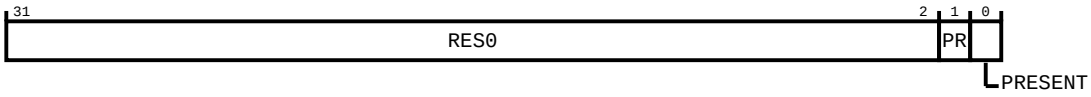
Reset value

0x0000000-

Bit descriptions

The following figure shows the SYSPCR1 register bit assignments.

Figure 9-122: Bit assignment diagram for the SYSPCR1 register



The following table shows the SYSPCR1 register bit descriptions.

Table 9-127: SYSPCR1 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	PR	0b0	RW	Power request. Reserved if SYSPCR1.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **0b0** Indicates that power is not requested for system domain 1 **0b1** Indicates that power is requested for system domain 1
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for system domain 1 **0b0** Indicates that the power request is not implemented for system domain 1 **0b1** Indicates that the power request is implemented for system domain 1

9.7.15 css600_apbrom_gpr_32bit System Power Control Register 2, SYSPCR2

The SYSPCR2 register indicates whether power has been requested for system domain 2.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xB08

Type

RW

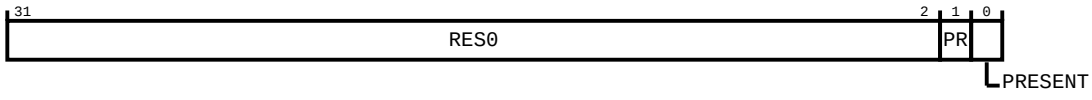
Reset value

0x0000000-

Bit descriptions

The following figure shows the SYSPCR2 register bit assignments.

Figure 9-123: Bit assignment diagram for the SYSPCR2 register



The following table shows the SYSPCR2 register bit descriptions.

Table 9-128: SYSPCR2 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	PR	0b0	RW	Power request. Reserved if SYSPCR2.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **0b0** Indicates that power is not requested for system domain 2 **0b1** Indicates that power is requested for system domain 2
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for system domain 2 **0b0** Indicates that the power request is not implemented for system domain 2 **0b1** Indicates that the power request is implemented for system domain 2

9.7.16 css600_apbrom_gpr_32bit System Power Control Register 31, SYSPCR31

The SYSPCR31 register indicates whether power has been requested for system domain 31.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xB7C

Type

RW

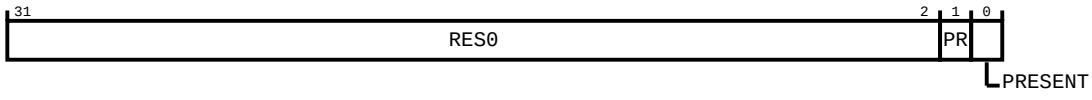
Reset value

0x0000000-

Bit descriptions

The following figure shows the SYSPCR31 register bit assignments.

Figure 9-124: Bit assignment diagram for the SYSPCR31 register



The following table shows the SYSPCR31 register bit descriptions.

Table 9-129: SYSPCR31 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	PR	0b0	RW	Power request. Reserved if SYSPCR31.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **0b0** Indicates that power is not requested for system domain 31 **0b1** Indicates that power is requested for system domain 31
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for system domain 31 **0b0** Indicates that the power request is not implemented for system domain 31 **0b1** Indicates that the power request is implemented for system domain 31

9.7.17 css600_apbrom_gpr_32bit System Power Status Register 0, SYSPSR0

The SYSPSR0 register indicates the power status for system domain 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xB80

Type

RO

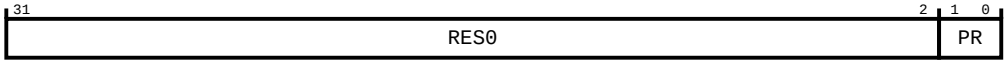
Reset value

0x0000000-

Bit descriptions

The following figure shows the SYSPSR0 register bit assignments.

Figure 9-125: Bit assignment diagram for the SYSPSR0 register



The following table shows the SYSPSR0 register bit descriptions.

Table 9-130: SYSPSR0 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	**UNKNOWN**	RO	Power status: **0b00** System domain 0 might not be powered **0b11** System domain 0 is powered and must remain powered until SYSPCR0.PR == 0

9.7.18 css600_apbrom_gpr_32bit System Power Status Register 1, SYSPSR1

The SYSPSR1 register indicates the power status for system domain 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xB84

Type

RO

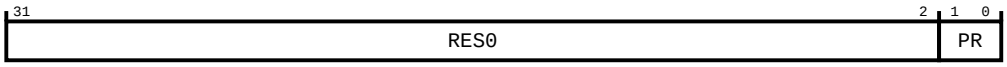
Reset value

0x0000000-

Bit descriptions

The following figure shows the SYSPSR1 register bit assignments.

Figure 9-126: Bit assignment diagram for the SYSPSR1 register



The following table shows the SYSPSR1 register bit descriptions.

Table 9-131: SYSPSR1 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
1:0	PR	UNKNOWN	RO	Power status: **0b00** System domain 1 might not be powered **0b11** System domain 1 is powered and must remain powered until SYSPCR1.PR == 0

9.7.19 css600_apbrom_gpr_32bit System Power Status Register 2, SYSPSR2

The SYSPSR2 register indicates the power status for system domain 2.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xB88

Type

RO

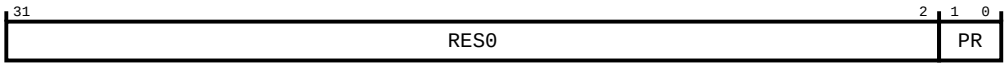
Reset value

0x0000000-

Bit descriptions

The following figure shows the SYSPSR2 register bit assignments.

Figure 9-127: Bit assignment diagram for the SYSPSR2 register



The following table shows the SYSPSR2 register bit descriptions.

Table 9-132: SYSPSR2 bit descriptions

Bits	Name	Reset	Type	Description
31:2	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	UNKNOWN	RO	Power status: **0b00** System domain 2 might not be powered **0b11** System domain 2 is powered and must remain powered until SYSPCR2.PR == 0

9.7.20 css600_apbrom_gpr_32bit System Power Status Register 31, SYSPSR31

The SYSPSR31 register indicates the power status for system domain 31.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xBFC`

Type

RO

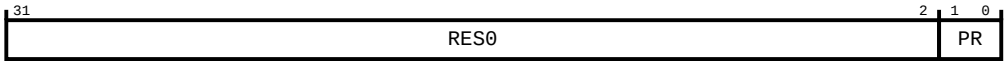
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the SYSPSR31 register bit assignments.

Figure 9-128: Bit assignment diagram for the SYSPSR31 register



The following table shows the SYSPSR31 register bit descriptions.

Table 9-133: SYSPSR31 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	**UNKNOWN**	RO	Power status: **0b00** System domain 31 might not be powered **0b11** System domain 31 is powered and must remain powered until SYSPCR31.PR == 0

9.7.21 css600_apbrom_gpr_32bit Power Request ID Register, PRIDR0

The PRIDR0 register indicates the version of the power request function.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC00

Type

RO

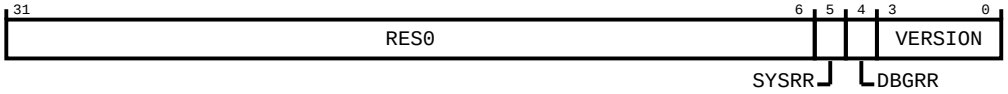
Reset value

0x00000031

Bit descriptions

The following figure shows the PRIDR0 register bit assignments.

Figure 9-129: Bit assignment diagram for the PRIDR0 register



The following table shows the PRIDR0 register bit descriptions.

Table 9-134: PRIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:6	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
5	SYSRR	0b1	RO	Indicates whether the system reset request functionality is present **0b1** System reset request functionality is implemented. SYSRSTRR and SYSRSTAR are both implemented.
4	DBGRR	0b1	RO	Indicates whether the debug reset request functionality is present: **0b1** Debug reset request functionality is implemented. DBGRSTRR and DBGRSTAR are both implemented.
3:0	VERSION	0b0001	RO	Version of the power request function. **0b0001** Power request functionality version 1 is included

9.7.22 css600_apbrom_gpr_32bit Debug Reset Request Register, DBGRSTRR

The DBGRSTRR register indicates the status of a debug reset request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC10

Type

RW

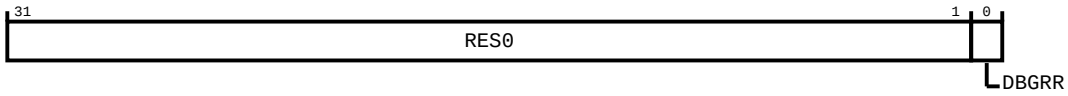
Reset value

0x00000000

Bit descriptions

The following figure shows the DBGRSTRR register bit assignments.

Figure 9-130: Bit assignment diagram for the DBGRSTRR register



The following table shows the DBGRSTRR register bit descriptions.

Table 9-135: DBGRSTRR bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	DBGRR	0b0	RW	Debug reset request. Software must clear this bit after setting it. There is no automatic mechanism to clear it. When Realm Management Extension is disabled (legacy_tz_en==1) software can set this bit when Secure invasive debug is enabled. When Realm Management Extension is enabled (legacy_tz_en==0) software can set this bit when Root Invasive Debug is enabled. **0b0** No reset is requested. cdbgrstreq output is LOW. **0b1** Reset is requested. cdbgrstreq output is HIGH.

9.7.23 css600_apbrom_gpr_32bit Debug Reset Acknowledge Register, DBGRSTAR

The DBGRSTAR register acknowledges a debug reset request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC14

Type

RO

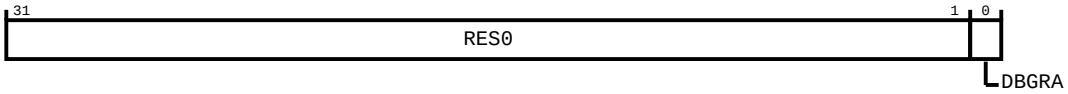
Reset value

0x00000000

Bit descriptions

The following figure shows the DBGRSTAR register bit assignments.

Figure 9-131: Bit assignment diagram for the DBGRSTAR register



The following table shows the DBGRSTAR register bit descriptions.

Table 9-136: DBGRSTAR bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	DBGRA	0b0	RO	Debug reset acknowledge: **0b0** No reset is requested or reset is not acknowledged **0b1** Reset is acknowledged by the external reset controller

9.7.24 css600_apbrom_gpr_32bit System Reset Request Register, SYSRSTRR

The SYSRSTRR register indicates the status of a system reset request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC18

Type

RW

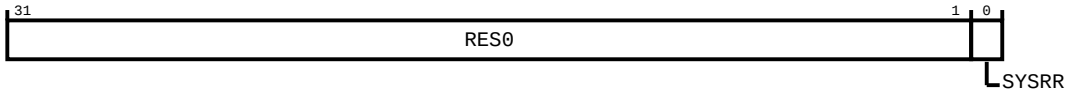
Reset value

0x00000000

Bit descriptions

The following figure shows the SYSRSTRR register bit assignments.

Figure 9-132: Bit assignment diagram for the SYSRSTRR register



The following table shows the SYSRSTRR register bit descriptions.

Table 9-137: SYSRSTRR bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	SYSRR	0b0	RW	System reset request. The software needs to clear this bit after setting it. There is no automatic mechanism to clear it. When Realm Management Extension is disabled (legacy_tz_en==1) software can set this bit when Secure invasive debug is enabled. When Realm Management Extension is enabled (legacy_tz_en==0) software can set this bit when Root Invasive Debug is enabled. **0b0** No reset is requested. csysrstreq output is LOW. **0b1** Reset is requested. csysrstreq output is HIGH.

9.7.25 css600_apbrom_gpr_32bit System Reset Acknowledge Register, SYSRSTAR

The SYSRSTAR register acknowledges a system reset request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC1C

Type

RO

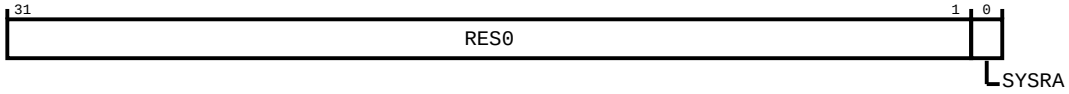
Reset value

0x00000000

Bit descriptions

The following figure shows the SYSRSTAR register bit assignments.

Figure 9-133: Bit assignment diagram for the SYSRSTAR register



The following table shows the SYSRSTAR register bit descriptions.

Table 9-138: SYSRSTAR bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	SYSRA	0b0	RO	System reset acknowledge: **0b0** No reset is requested or reset is not acknowledged **0b1** Reset is acknowledged by the external reset controller

9.7.26 css600_apbrom_gpr_32bit Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

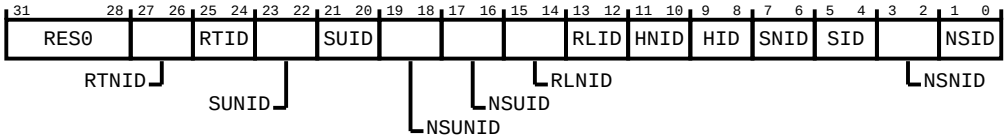
Reset value

0x0-00-0--

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-134: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-139: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	**UNKNOWN**	RO	Root non-invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
25:24	RTID	**UNKNOWN**	RO	Root invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	**UNKNOWN**	RO	Realm non-invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.

Bits	Name	Reset	Type	Description
13:12	RLID	**UNKNOWN**	RO	Realm invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	**UNKNOWN**	RO	Secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
5:4	SID	**UNKNOWN**	RO	Secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
3:2	NSNID	**UNKNOWN**	RO	Non-secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
1:0	NSID	**UNKNOWN**	RO	Non-secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.

9.7.27 css600_apbrom_gpr_32bit Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFBC

Type

RO

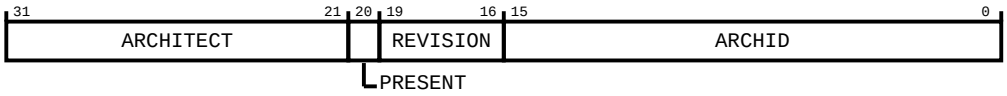
Reset value

0x47700AF7

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-135: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-140: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x23B	RO	Defines the architect of the component **0x23B** Arm
20	PRESENT	0b1	RO	Indicates the presence of this register **0b1** DEVARCH is present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0AF7	RO	Architecture ID. Returns a value that identifies the architecture of the component. **0x0AF7** CoreSight ROM architecture

9.7.28 css600_apbrom_gpr_32bit Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC8

Type

RO

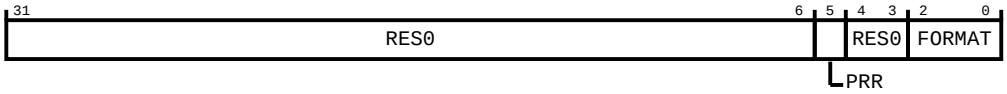
Reset value

0x0000002-

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-136: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-141: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:6	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
5	PRR	0b1	RO	Indicates that power request functionality is included. **0b1** GPR is included
4:3	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
2:0	FORMAT	**IMPLEMENTATION DEFINED**	RO	ROM format. **0b000** 32-bit format **0b001** 64-bit format

9.7.29 css600_apbrom_gpr_32bit Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

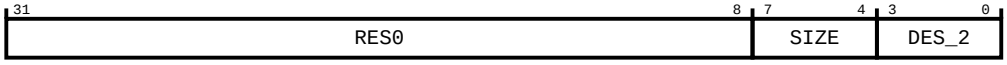
Reset value

0x0000000-

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-137: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-142: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	**IMPLEMENTATION DEFINED**	RO	JEP106 continuation code, bits [3:0]. Set by the configuration inputs jep106_if[3:0]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.7.30 css600_apbrom_gpr_32bit Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

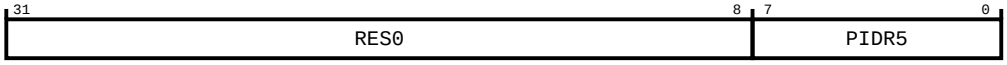
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-138: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-143: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.7.31 css600_apbrom_gpr_32bit Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD8

Type

RO

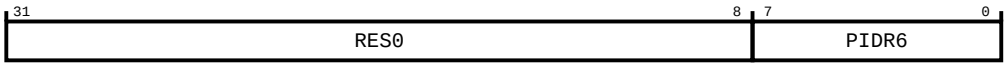
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-139: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-144: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.7.32 css600_apbrom_gpr_32bit Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

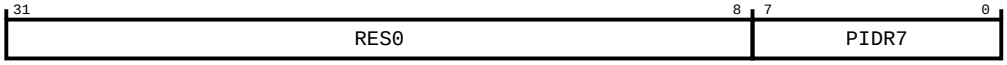
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-140: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-145: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.7.33 css600_apbrom_gpr_32bit Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFE0`

Type

RO

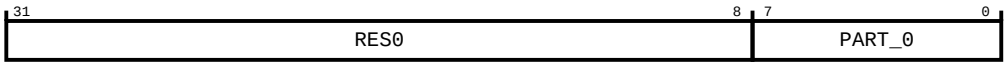
Reset value

`0x000000--`

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-141: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-146: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	**IMPLEMENTATION DEFINED**	RO	Part number, bits[7:0]. Set by the configuration inputs part_number[7:0]

9.7.34 css600_apbrom_gpr_32bit Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

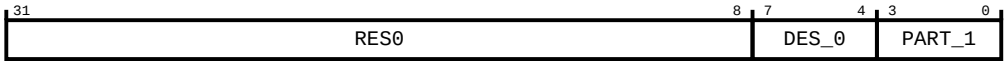
Reset value

0x000000--

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-142: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-147: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	**IMPLEMENTATION DEFINED**	RO	JEP106 identification code, bits[3:0]. Set by the configuration inputs jep106_id[3:0]. Together, with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	**IMPLEMENTATION DEFINED**	RO	Part number, bits[11:8]. Set by the configuration inputs part_number[11:8].

9.7.35 css600_apbrom_gpr_32bit Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

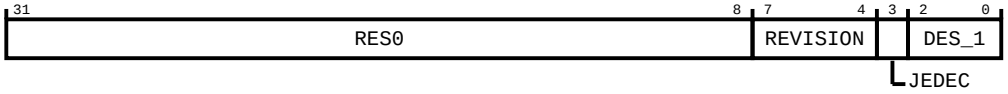
Reset value

0x000000--

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-143: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-148: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	**IMPLEMENTATION DEFINED**	RO	Revision. Set by the configuration inputs revision[3:0].
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	**IMPLEMENTATION DEFINED**	RO	JEP106 identification code, bits[6:4]. Set by the configuration inputs jep106_id[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.7.36 css600_apbrom_gpr_32bit Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFEC

Type

RO

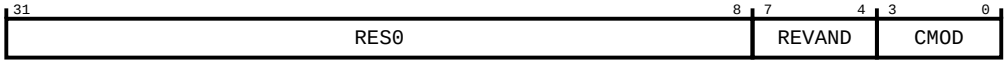
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-144: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-149: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.7.37 css600_apbrom_gpr_32bit Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

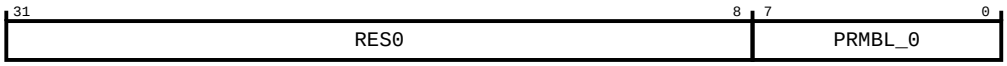
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-145: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-150: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.7.38 css600_apbrom_gpr_32bit Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

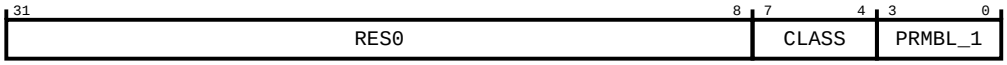
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-146: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-151: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.7.39 css600_apbrom_gpr_32bit Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF8`

Type

RO

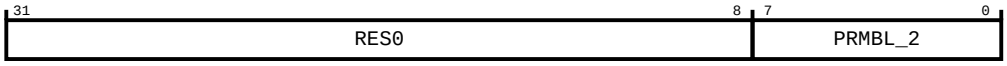
Reset value

`0x00000005`

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-147: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-152: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.7.40 css600_apbrom_gpr_32bit Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

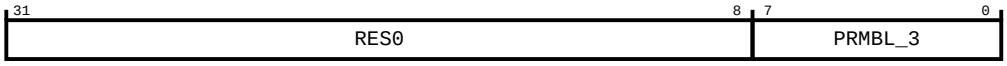
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-148: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-153: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.8 css600_apbrom_gpr_64bit register summary

This section describes the css600_apbrom_gpr_64bit_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-154: css600_apbrom_gpr_64bit_registers register summary

Offset	Name	Type	Reset	Width	Description
0x0	ROMEntry0	RO	0x--------	32-bit	ROM Entries register 0
0x4	ROMEntry0HI	RO	0x--------	32-bit	ROM Entries register 0 bits [63:32]
0x8	ROMEntry1	RO	0x--------	32-bit	ROM Entries register 1
0xc	ROMEntry1HI	RO	0x--------	32-bit	ROM Entries register 1 bits [63:32]
0x10	ROMEntry2	RO	0x--------	32-bit	ROM Entries register 2
0x14	ROMEntry2HI	RO	0x--------	32-bit	ROM Entries register 2 bits [63:32]
...	...				
0x7f8	ROMEntry255	RO	0x--------	32-bit	ROM Entries register 255
0x7fc	ROMEntry255HI	RO	0x--------	32-bit	ROM Entries register 255 bits [63:32]
0xa00	DBGPCR0	RW	0x00000001	32-bit	Debug Power Control Register 0
0xa04	DBGPCR1	RW	0x0000000-	32-bit	Debug Power Control Register 1

Offset	Name	Type	Reset	Width	Description
0xa08	DBGPCR2	RW	0x0000000-	32-bit	Debug Power Control Register 2
0xa7c	DBGPCR31	RW	0x0000000-	32-bit	Debug Power Control Register 31
0xa80	DBGPSR0	RO	0x0000000-	32-bit	Debug Power Status Register 0
0xa84	DBGPSR1	RO	0x0000000-	32-bit	Debug Power Status Register 1
0xa88	DBGPSR2	RO	0x0000000-	32-bit	Debug Power Status Register 2
...	...				
0xafc	DBGPSR31	RO	0x0000000-	32-bit	Debug Power Status Register 31
0xb00	SYSPCR0	RW	0x00000001	32-bit	System Power Control Register 0
0xb04	SYSPCR1	RW	0x0000000-	32-bit	System Power Control Register 1
0xb08	SYSPCR2	RW	0x0000000-	32-bit	System Power Control Register 2
...	...				
0xb7c	SYSPCR31	RW	0x0000000-	32-bit	System Power Control Register 31
0xb80	SYSPSR0	RO	0x0000000-	32-bit	System Power Status Register 0
0xb84	SYSPSR1	RO	0x0000000-	32-bit	System Power Status Register 1
0xb88	SYSPSR2	RO	0x0000000-	32-bit	System Power Status Register 2
...	...				
0xbfc	SYSPSR31	RO	0x0000000-	32-bit	System Power Status Register 31
0xc00	PRIDR0	RO	0x00000031	32-bit	Power Request ID Register
0xc10	DBGRSTRR	RW	0x00000000	32-bit	Debug Reset Request Register
0xc14	DBGRSTAR	RO	0x00000000	32-bit	Debug Reset Acknowledge Register
0xc18	SYSRSTRR	RW	0x00000000	32-bit	System Reset Request Register
0xc1c	SYSRSTAR	RO	0x00000000	32-bit	System Reset Acknowledge Register
0xfb8	AUTHSTATUS	RO	0x0-00-0--	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x47700AF7	32-bit	Device Architecture Register
0xfc8	DEVID	RO	0x0000002-	32-bit	Device Configuration Register
0xfd0	PIDR4	RO	0x0000000-	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000--	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000--	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x000000--	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.8.1 css600_apbrom_gpr_64bit ROM Entries register 0, ROMEntry0

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x0`

Type

RO

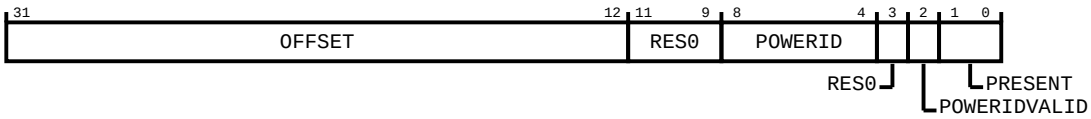
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry0 register bit assignments.

Figure 9-149: Bit assignment diagram for the ROMEntry0 register



The following table shows the ROMEntry0 register bit descriptions.

Table 9-155: ROMEntry0 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **0b0** POWERID field of this register is not valid **0b1** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **0b00** ROM table entry not present. This is the last entry. **0b01** Reserved **0b10** ROM table entry not present. This is not the last entry. **0b11** ROM table entry present

9.8.2 css600_apbrom_gpr_64bit ROM Entries register 0 bits [63:32], ROMEntry0HI

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x4`

Type

RO

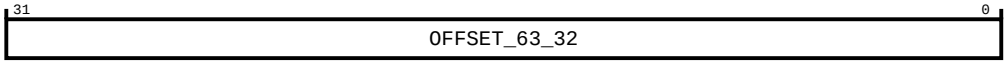
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry0HI register bit assignments.

Figure 9-150: Bit assignment diagram for the ROMEntry0HI register



The following table shows the ROMEntry0HI register bit descriptions.

Table 9-156: ROMEntry0HI bit descriptions

Bits	Name	Reset	Type	Description
31:0	OFFSET_63_32	**IMPLEMENTATION DEFINED**	RO	Upper 32 bits of the component address offset.

9.8.3 css600_apbrom_gpr_64bit ROM Entries register 1, ROMEntry1

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x8`

Type

RO

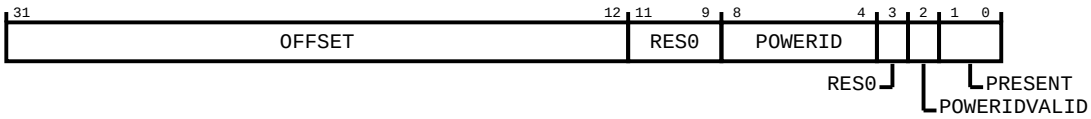
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry1 register bit assignments.

Figure 9-151: Bit assignment diagram for the ROMEntry1 register



The following table shows the ROMEntry1 register bit descriptions.

Table 9-157: ROMEntry1 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **`0b0`** POWERID field of this register is not valid **`0b1`** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **`0b00`** ROM table entry not present. This is the last entry. **`0b01`** Reserved **`0b10`** ROM table entry not present. This is not the last entry. **`0b11`** ROM table entry present

9.8.4 css600_apbrom_gpr_64bit ROM Entries register 1 bits [63:32], ROMEntry1HI

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xC`

Type

RO

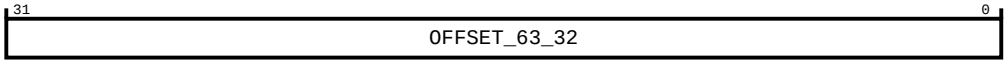
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry1HI register bit assignments.

Figure 9-152: Bit assignment diagram for the ROMEntry1HI register



The following table shows the ROMEntry1HI register bit descriptions.

Table 9-158: ROMEntry1HI bit descriptions

Bits	Name	Reset	Type	Description
31:0	OFFSET_63_32	IMPLEMENTATION DEFINED	RO	Upper 32 bits of the component address offset.

9.8.5 css600_apbrom_gpr_64bit ROM Entries register 2, ROMEntry2

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x10`

Type

RO

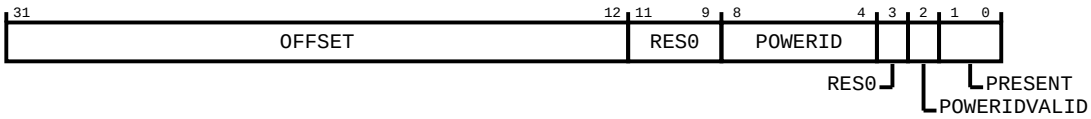
Reset value

`0x--------`

Bit descriptions

The following figure shows the ROMEntry2 register bit assignments.

Figure 9-153: Bit assignment diagram for the ROMEntry2 register



The following table shows the ROMEntry2 register bit descriptions.

Table 9-159: ROMEntry2 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **0b0** POWERID field of this register is not valid **0b1** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **0b00** ROM table entry not present. This is the last entry. **0b01** Reserved **0b10** ROM table entry not present. This is not the last entry. **0b11** ROM table entry present

9.8.6 css600_apbrom_gpr_64bit ROM Entries register 2 bits [63:32], ROMEntry2HI

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x14`

Type

RO

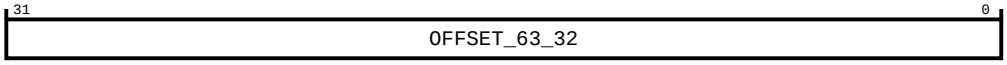
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry2HI register bit assignments.

Figure 9-154: Bit assignment diagram for the ROMEntry2HI register



The following table shows the ROMEntry2HI register bit descriptions.

Table 9-160: ROMEntry2HI bit descriptions

Bits	Name	Reset	Type	Description
31:0	OFFSET_63_32	IMPLEMENTATION DEFINED	RO	Upper 32 bits of the component address offset.

9.8.7 css600_apbrom_gpr_64bit ROM Entries register 255, ROMEntry255

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x7F8

Type

RO

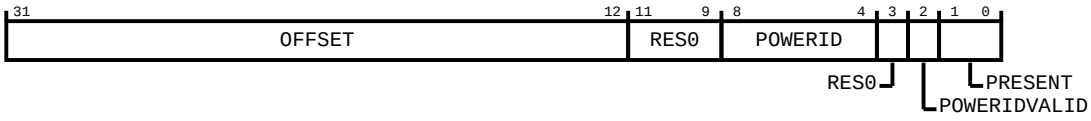
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry255 register bit assignments.

Figure 9-155: Bit assignment diagram for the ROMEntry255 register



The following table shows the ROMEntry255 register bit descriptions.

Table 9-161: ROMEntry255 bit descriptions

Bits	Name	Reset	Type	Description
31:12	OFFSET	**IMPLEMENTATION DEFINED**	RO	The component address, relative to the base address of this ROM table. The component address is calculated using the following equation: Component Address = ROM Table Base Address + (OFFSET << 12). If a component occupies more than a single 4KB block, OFFSET points to the 4KB block which contains the Peripheral ID and Component ID registers for the component. Negative values of OFFSET are permitted, using two's complement.
11:9	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
8:4	POWERID	**IMPLEMENTATION DEFINED**	RO	Indicates the power domain ID of the component. Only valid if bit 2 is set. If bit 2 is clear then this field has a value of 0. Possible values are 0 to 31, representing the 32 DBGPWRUPREQ/ACK interface pins of the component.
3	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
2	POWERIDVALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether there is a power domain ID specified in the ROM table entry: **`0b0`** POWERID field of this register is not valid **`0b1`** POWERID field of this register is valid
1:0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates whether the ROM table entry is present: **`0b00`** ROM table entry not present. This is the last entry. **`0b01`** Reserved **`0b10`** ROM table entry not present. This is not the last entry. **`0b11`** ROM table entry present

9.8.8 css600_apbrom_gpr_64bit ROM Entries register 255 bits [63:32], ROMEntry255HI

The ROMEntry register contains a descriptor of a CoreSight component in the system. All ROM table entries conform to the same format.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x7FC`

Type

RO

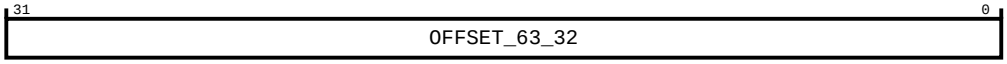
Reset value

0x--------

Bit descriptions

The following figure shows the ROMEntry255HI register bit assignments.

Figure 9-156: Bit assignment diagram for the ROMEntry255HI register



The following table shows the ROMEntry255HI register bit descriptions.

Table 9-162: ROMEntry255HI bit descriptions

Bits	Name	Reset	Type	Description
31:0	OFFSET_63_32	IMPLEMENTATION DEFINED	RO	Upper 32 bits of the component address offset.

9.8.9 css600_apbrom_gpr_64bit Debug Power Control Register 0, DBGPCR0

The DBGPCR0 register indicates whether power has been requested for debug domain 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xA00

Type

RW

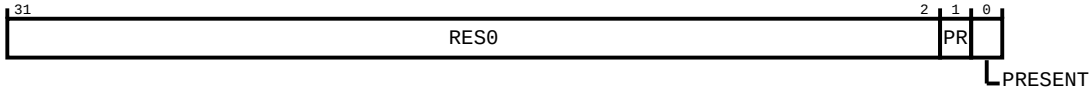
Reset value

0x00000001

Bit descriptions

The following figure shows the DBGPCR0 register bit assignments.

Figure 9-157: Bit assignment diagram for the DBGPCR0 register



The following table shows the DBGPCR0 register bit descriptions.

Table 9-163: DBGPCR0 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	PR	0b0	RW	Power request. Software can set this bit when non-invasive debug is enabled. **0b0** Indicates that power is not requested for debug domain 0 **0b1** Indicates that power is requested for debug domain 0
0	PRESENT	0b1	RO	Indicates the presence of power domain control for debug domain 0: **0b1** Indicates that the power request is implemented for debug domain 0

9.8.10 css600_apbrom_gpr_64bit Debug Power Control Register 1, DBGPCR1

The DBGPCR1 register indicates whether power has been requested for debug domain 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xA04

Type

RW

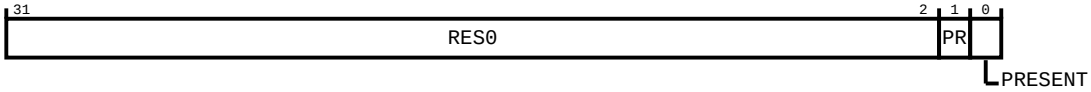
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPCR1 register bit assignments.

Figure 9-158: Bit assignment diagram for the DBGPCR1 register



The following table shows the DBGPCR1 register bit descriptions.

Table 9-164: DBGPCR1 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	PR	0b0	RW	Power request. Reserved if DBGPCR1.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **0b0** Indicates that power is not requested for debug domain 1 **0b1** Indicates that power is requested for debug domain 1
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for debug domain 1: **0b0** Indicates that the power request is not implemented for debug domain 1 **0b1** Indicates that the power request is implemented for debug domain 1

9.8.11 css600_apbrom_gpr_64bit Debug Power Control Register 2, DBGPCR2

The DBGPCR2 register indicates whether power has been requested for debug domain 2.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xA08

Type

RW

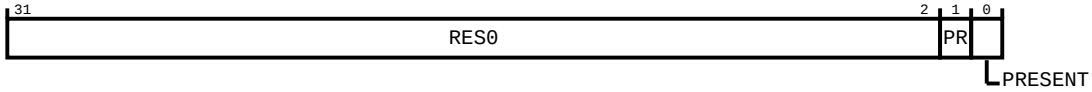
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPCR2 register bit assignments.

Figure 9-159: Bit assignment diagram for the DBGPCR2 register



The following table shows the DBGPCR2 register bit descriptions.

Table 9-165: DBGPCR2 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	PR	0b0	RW	Power request. Reserved if DBGPCR2.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **0b0** Indicates that power is not requested for debug domain 2 **0b1** Indicates that power is requested for debug domain 2
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for debug domain 2: **0b0** Indicates that the power request is not implemented for debug domain 2 **0b1** Indicates that the power request is implemented for debug domain 2

9.8.12 css600_apbrom_gpr_64bit Debug Power Control Register 31, DBGPCR31

The DBGPCR31 register indicates whether power has been requested for debug domain 31.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xA7C

Type

RW

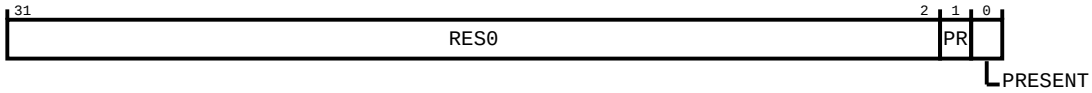
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPCR31 register bit assignments.

Figure 9-160: Bit assignment diagram for the DBGPCR31 register



The following table shows the DBGPCR31 register bit descriptions.

Table 9-166: DBGPCR31 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	PR	0b0	RW	Power request. Reserved if DBGPCR31.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **0b0** Indicates that power is not requested for debug domain 31 **0b1** Indicates that power is requested for debug domain 31
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for debug domain 31: **0b0** Indicates that the power request is not implemented for debug domain 31 **0b1** Indicates that the power request is implemented for debug domain 31

9.8.13 css600_apbrom_gpr_64bit Debug Power Status Register 0, DBGPSR0

The DBGPSR0 register indicates the power status for debug domain 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xA80

Type

RO

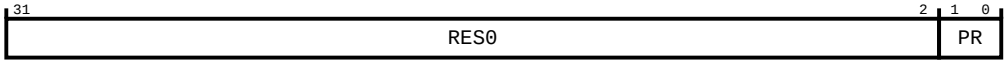
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPSR0 register bit assignments.

Figure 9-161: Bit assignment diagram for the DBGPSR0 register



The following table shows the DBGPSR0 register bit descriptions.

Table 9-167: DBGPSR0 bit descriptions

Bits	Name	Reset	Type	Description
31:2	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	UNKNOWN	RO	Power status. **0b00** Debug domain 0 might not be powered. **0b11** Debug domain 0 is powered and must remain powered until DBGPCR0.PR == 0.

9.8.14 css600_apbrom_gpr_64bit Debug Power Status Register 1, DBGPSR1

The DBGPSR1 register indicates the power status for debug domain 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xA84

Type

RO

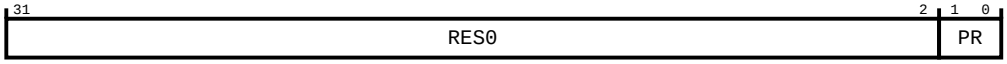
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPSR1 register bit assignments.

Figure 9-162: Bit assignment diagram for the DBGPSR1 register



The following table shows the DBGPSR1 register bit descriptions.

Table 9-168: DBGPSR1 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	**UNKNOWN**	RO	Power status. **0b00** Debug domain 1 might not be powered. **0b11** Debug domain 1 is powered and must remain powered until DBGPCR1.PR == 0.

9.8.15 css600_apbrom_gpr_64bit Debug Power Status Register 2, DBGPSR2

The DBGPSR2 register indicates the power status for debug domain 2.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xA88

Type

RO

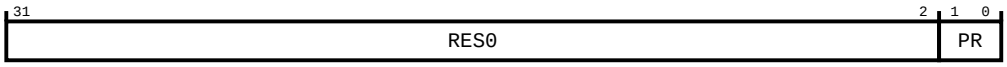
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPSR2 register bit assignments.

Figure 9-163: Bit assignment diagram for the DBGPSR2 register



The following table shows the DBGPSR2 register bit descriptions.

Table 9-169: DBGPSR2 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
1:0	PR	UNKNOWN	RO	Power status. **0b00** Debug domain 2 might not be powered. **0b11** Debug domain 2 is powered and must remain powered until DBGPCR2.PR == 0.

9.8.16 css600_apbrom_gpr_64bit Debug Power Status Register 31, DBGPSR31

The DBGPSR31 register indicates the power status for debug domain 31.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xAFC

Type

RO

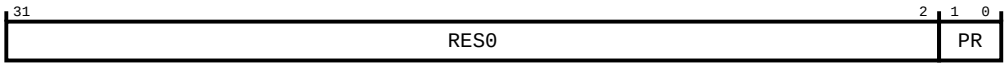
Reset value

0x0000000-

Bit descriptions

The following figure shows the DBGPSR31 register bit assignments.

Figure 9-164: Bit assignment diagram for the DBGPSR31 register



The following table shows the DBGPSR31 register bit descriptions.

Table 9-170: DBGPSR31 bit descriptions

Bits	Name	Reset	Type	Description
31:2	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	UNKNOWN	RO	Power status. **0b00** Debug domain 31 might not be powered. **0b11** Debug domain 31 is powered and must remain powered until DBGPCR31.PR == 0.

9.8.17 css600_apbrom_gpr_64bit System Power Control Register 0, SYSPCR0

The SYSPCR0 register indicates whether power has been requested for system domain 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xB00`

Type

RW

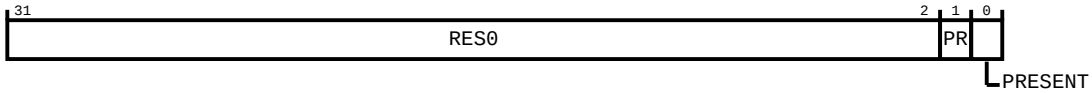
Reset value

`0x00000001`

Bit descriptions

The following figure shows the SYSPCR0 register bit assignments.

Figure 9-165: Bit assignment diagram for the SYSPCR0 register



The following table shows the SYSPCR0 register bit descriptions.

Table 9-171: SYSPCR0 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1	PR	`0b0`	RW	Power request. Software can set this bit when non-invasive debug is enabled. **`0b0`** Indicates that power is not requested for system domain 0 **`0b1`** Indicates that power is requested for system domain 0
0	PRESENT	`0b1`	RO	Indicates the presence of power domain control for system domain 0 **`0b1`** Indicates that the power request is implemented for system domain 0

9.8.18 css600_apbrom_gpr_64bit System Power Control Register 1, SYSPCR1

The SYSPCR1 register indicates whether power has been requested for system domain 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xB04`

Type

RW

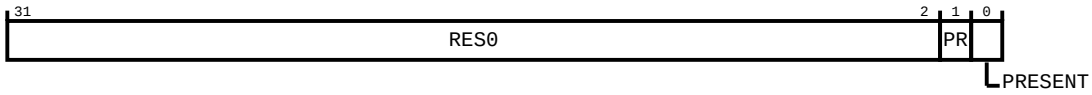
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the SYSPCR1 register bit assignments.

Figure 9-166: Bit assignment diagram for the SYSPCR1 register



The following table shows the SYSPCR1 register bit descriptions.

Table 9-172: SYSPCR1 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1	PR	`0b0`	RW	Power request. Reserved if SYSPCR1.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **`0b0`** Indicates that power is not requested for system domain 1 **`0b1`** Indicates that power is requested for system domain 1
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for system domain 1 **`0b0`** Indicates that the power request is not implemented for system domain 1 **`0b1`** Indicates that the power request is implemented for system domain 1

9.8.19 css600_apbrom_gpr_64bit System Power Control Register 2, SYSPCR2

The SYSPCR2 register indicates whether power has been requested for system domain 2.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xB08`

Type

RW

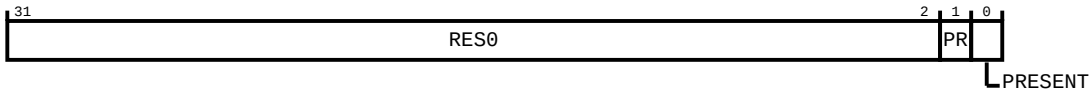
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the SYSPCR2 register bit assignments.

Figure 9-167: Bit assignment diagram for the SYSPCR2 register



The following table shows the SYSPCR2 register bit descriptions.

Table 9-173: SYSPCR2 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1	PR	`0b0`	RW	Power request. Reserved if SYSPCR2.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **`0b0`** Indicates that power is not requested for system domain 2 **`0b1`** Indicates that power is requested for system domain 2
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for system domain 2 **`0b0`** Indicates that the power request is not implemented for system domain 2 **`0b1`** Indicates that the power request is implemented for system domain 2

9.8.20 css600_apbrom_gpr_64bit System Power Control Register 31, SYSPCR31

The SYSPCR31 register indicates whether power has been requested for system domain 31.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xB7C

Type

RW

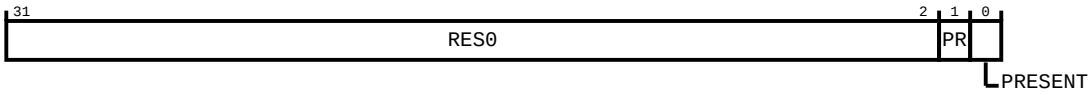
Reset value

0x0000000-

Bit descriptions

The following figure shows the SYSPCR31 register bit assignments.

Figure 9-168: Bit assignment diagram for the SYSPCR31 register



The following table shows the SYSPCR31 register bit descriptions.

Table 9-174: SYSPCR31 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	PR	0b0	RW	Power request. Reserved if SYSPCR31.PRESENT == 0. Software can set this bit when non-invasive debug is enabled. **0b0** Indicates that power is not requested for system domain 31 **0b1** Indicates that power is requested for system domain 31
0	PRESENT	**IMPLEMENTATION DEFINED**	RO	Indicates the presence of power domain control for system domain 31 **0b0** Indicates that the power request is not implemented for system domain 31 **0b1** Indicates that the power request is implemented for system domain 31

9.8.21 css600_apbrom_gpr_64bit System Power Status Register 0, SYSPSR0

The SYSPSR0 register indicates the power status for system domain 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xB80`

Type

RO

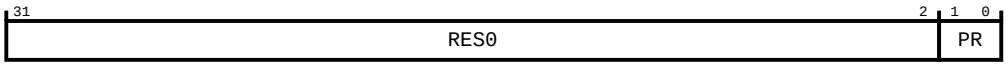
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the SYSPSR0 register bit assignments.

Figure 9-169: Bit assignment diagram for the SYSPSR0 register



The following table shows the SYSPSR0 register bit descriptions.

Table 9-175: SYSPSR0 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1:0	PR	**UNKNOWN**	RO	Power status: **`0b00`** System domain 0 might not be powered **`0b11`** System domain 0 is powered and must remain powered until SYSPCR0.PR == 0

9.8.22 css600_apbrom_gpr_64bit System Power Status Register 1, SYSPSR1

The SYSPSR1 register indicates the power status for system domain 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xB84

Type

RO

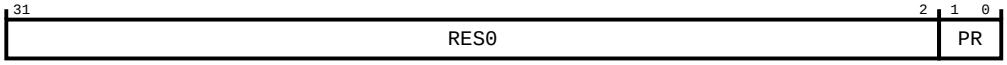
Reset value

0x0000000-

Bit descriptions

The following figure shows the SYSPSR1 register bit assignments.

Figure 9-170: Bit assignment diagram for the SYSPSR1 register



The following table shows the SYSPSR1 register bit descriptions.

Table 9-176: SYSPSR1 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	**UNKNOWN**	RO	Power status: **0b00** System domain 1 might not be powered **0b11** System domain 1 is powered and must remain powered until SYSPCR1.PR == 0

9.8.23 css600_apbrom_gpr_64bit System Power Status Register 2, SYSPSR2

The SYSPSR2 register indicates the power status for system domain 2.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xB88

Type

RO

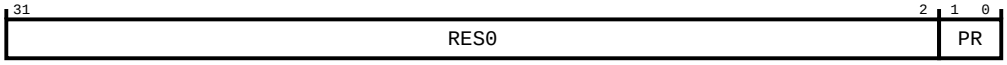
Reset value

0x0000000-

Bit descriptions

The following figure shows the SYSPSR2 register bit assignments.

Figure 9-171: Bit assignment diagram for the SYSPSR2 register



The following table shows the SYSPSR2 register bit descriptions.

Table 9-177: SYSPSR2 bit descriptions

Bits	Name	Reset	Type	Description
31:2	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	UNKNOWN	RO	Power status: **0b00** System domain 2 might not be powered **0b11** System domain 2 is powered and must remain powered until SYSPCR2.PR == 0

9.8.24 css600_apbrom_gpr_64bit System Power Status Register 31, SYSPSR31

The SYSPSR31 register indicates the power status for system domain 31.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xBFC

Type

RO

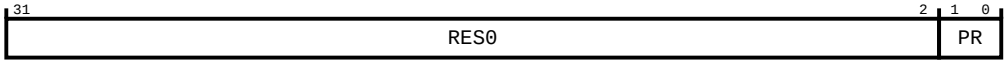
Reset value

0x0000000-

Bit descriptions

The following figure shows the SYSPSR31 register bit assignments.

Figure 9-172: Bit assignment diagram for the SYSPSR31 register



The following table shows the SYSPSR31 register bit descriptions.

Table 9-178: SYSPSR31 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1:0	PR	**UNKNOWN**	RO	Power status: **0b00** System domain 31 might not be powered **0b11** System domain 31 is powered and must remain powered until SYSPCR31.PR == 0

9.8.25 css600_apbrom_gpr_64bit Power Request ID Register, PRIDR0

The PRIDR0 register indicates the version of the power request function.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC00

Type

RO

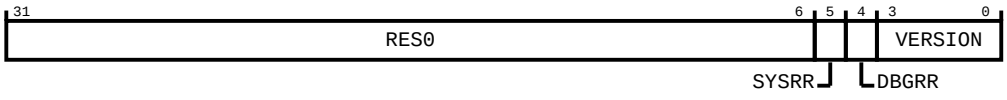
Reset value

0x00000031

Bit descriptions

The following figure shows the PRIDR0 register bit assignments.

Figure 9-173: Bit assignment diagram for the PRIDR0 register



The following table shows the PRIDR0 register bit descriptions.

Table 9-179: PRIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:6	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
5	SYSRR	0b1	RO	Indicates whether the system reset request functionality is present **0b1** System reset request functionality is implemented. SYSRSTRR and SYSRSTAR are both implemented.
4	DBGRR	0b1	RO	Indicates whether the debug reset request functionality is present: **0b1** Debug reset request functionality is implemented. DBGRSTRR and DBGRSTAR are both implemented.
3:0	VERSION	0b0001	RO	Version of the power request function. **0b0001** Power request functionality version 1 is included

9.8.26 css600_apbrom_gpr_64bit Debug Reset Request Register, DBGRSTRR

The DBGRSTRR register indicates the status of a debug reset request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC10

Type

RW

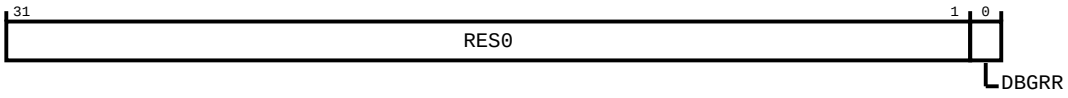
Reset value

0x00000000

Bit descriptions

The following figure shows the DBGRSTRR register bit assignments.

Figure 9-174: Bit assignment diagram for the DBGRSTRR register



The following table shows the DBGRSTRR register bit descriptions.

Table 9-180: DBGRSTRR bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	DBGRR	0b0	RW	Debug reset request. Software must clear this bit after setting it. There is no automatic mechanism to clear it. When Realm Management Extension is disabled (legacy_tz_en==1) software can set this bit when Secure invasive debug is enabled. When Realm Management Extension is enabled (legacy_tz_en==0) software can set this bit when Root Invasive Debug is enabled. **0b0** No reset is requested. cdbgrstreq output is LOW. **0b1** Reset is requested. cdbgrstreq output is HIGH.

9.8.27 css600_apbrom_gpr_64bit Debug Reset Acknowledge Register, DBGRSTAR

The DBGRSTAR register acknowledges a debug reset request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC14

Type

RO

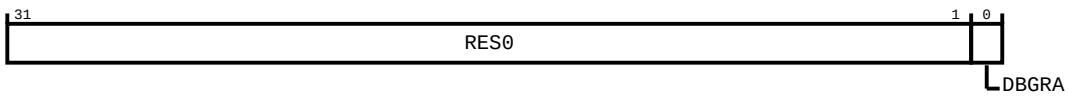
Reset value

0x00000000

Bit descriptions

The following figure shows the DBGRSTAR register bit assignments.

Figure 9-175: Bit assignment diagram for the DBGRSTAR register



The following table shows the DBGRSTAR register bit descriptions.

Table 9-181: DBGRSTAR bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
0	DBGRA	0b0	RO	Debug reset acknowledge: **0b0** No reset is requested or reset is not acknowledged **0b1** Reset is acknowledged by the external reset controller

9.8.28 css600_apbrom_gpr_64bit System Reset Request Register, SYSRSTRR

The SYSRSTRR register indicates the status of a system reset request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC18

Type

RW

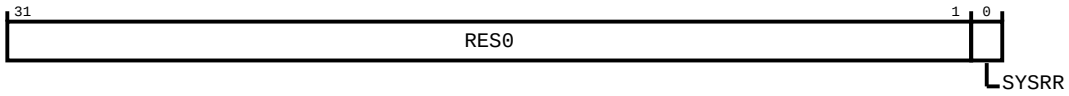
Reset value

0x00000000

Bit descriptions

The following figure shows the SYSRSTRR register bit assignments.

Figure 9-176: Bit assignment diagram for the SYSRSTRR register



The following table shows the SYSRSTRR register bit descriptions.

Table 9-182: SYSRSTRR bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
0	SYSRR	0b0	RW	System reset request. The software needs to clear this bit after setting it. There is no automatic mechanism to clear it. When Realm Management Extension is disabled (legacy_tz_en==1) software can set this bit when Secure invasive debug is enabled. When Realm Management Extension is enabled (legacy_tz_en==0) software can set this bit when Root Invasive Debug is enabled. **0b0** No reset is requested. csysrstreq output is LOW. **0b1** Reset is requested. csysrstreq output is HIGH.

9.8.29 css600_apbrom_gpr_64bit System Reset Acknowledge Register, SYSRSTAR

The SYSRSTAR register acknowledges a system reset request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC1C

Type

RO

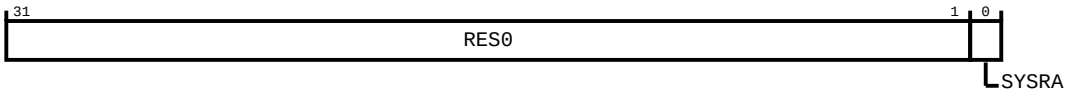
Reset value

0x00000000

Bit descriptions

The following figure shows the SYSRSTAR register bit assignments.

Figure 9-177: Bit assignment diagram for the SYSRSTAR register



The following table shows the SYSRSTAR register bit descriptions.

Table 9-183: SYSRSTAR bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
0	SYSRA	0b0	RO	System reset acknowledge: **0b0** No reset is requested or reset is not acknowledged **0b1** Reset is acknowledged by the external reset controller

9.8.30 css600_apbrom_gpr_64bit Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

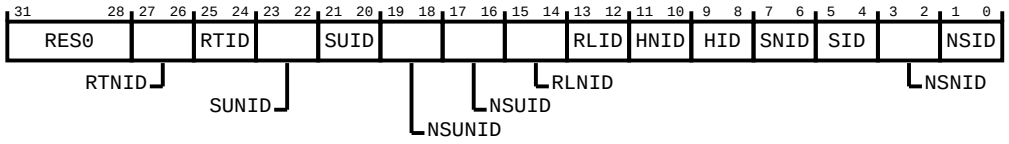
Reset value

0x0-00-0--

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-178: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-184: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	RES0	0b0000	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
27:26	RTNID	**UNKNOWN**	RO	Root non-invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
25:24	RTID	**UNKNOWN**	RO	Root invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	**UNKNOWN**	RO	Realm non-invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
13:12	RLID	**UNKNOWN**	RO	Realm invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
7:6	SNID	**UNKNOWN**	RO	Secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
5:4	SID	**UNKNOWN**	RO	Secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
3:2	NSNID	**UNKNOWN**	RO	Non-secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
1:0	NSID	**UNKNOWN**	RO	Non-secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.

9.8.31 css600_apbrom_gpr_64bit Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFBC`

Type

RO

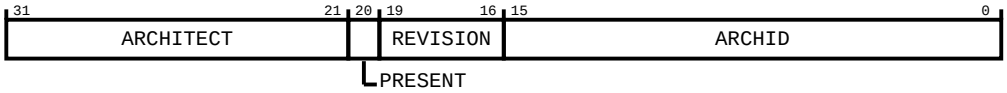
Reset value

`0x47700AF7`

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-179: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-185: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x23B	RO	Defines the architect of the component **0x23B** Arm
20	PRESENT	0b1	RO	Indicates the presence of this register **0b1** DEVARCH is present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0AF7	RO	Architecture ID. Returns a value that identifies the architecture of the component. **0x0AF7** CoreSight ROM architecture

9.8.32 css600_apbrom_gpr_64bit Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC8

Type

RO

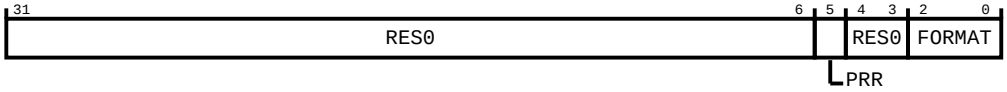
Reset value

0x0000002-

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-180: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-186: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:6	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
5	PRR	0b1	RO	Indicates that power request functionality is included. **0b1** GPR is included
4:3	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
2:0	FORMAT	**IMPLEMENTATION DEFINED**	RO	ROM format. **0b000** 32-bit format **0b001** 64-bit format

9.8.33 css600_apbrom_gpr_64bit Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

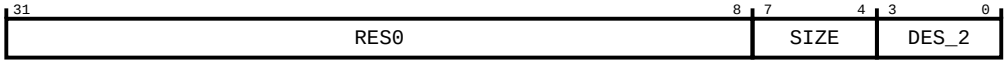
Reset value

0x0000000-

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-181: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-187: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	**IMPLEMENTATION DEFINED**	RO	JEP106 continuation code, bits [3:0]. Set by the configuration inputs jep106_if[3:0]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.8.34 css600_apbrom_gpr_64bit Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

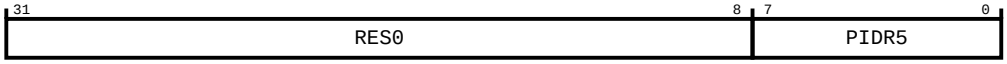
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-182: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-188: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.8.35 css600_apbrom_gpr_64bit Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD8

Type

RO

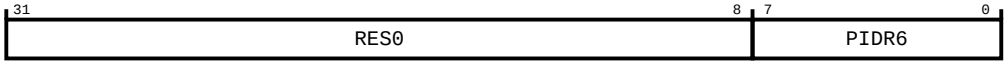
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-183: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-189: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.8.36 css600_apbrom_gpr_64bit Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFDC`

Type

RO

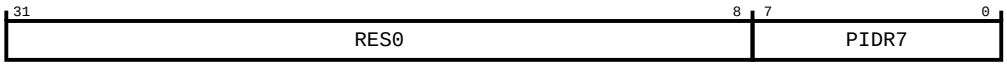
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-184: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-190: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.8.37 css600_apbrom_gpr_64bit Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

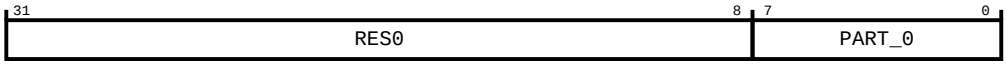
Reset value

0x000000--

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-185: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-191: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	IMPLEMENTATION DEFINED	RO	Part number, bits[7:0]. Set by the configuration inputs part_number[7:0]

9.8.38 css600_apbrom_gpr_64bit Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

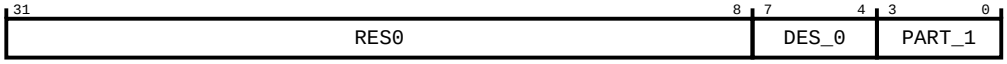
Reset value

0x000000--

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-186: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-192: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	IMPLEMENTATION DEFINED	RO	JEP106 identification code, bits[3:0]. Set by the configuration inputs jep106_id[3:0]. Together, with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	IMPLEMENTATION DEFINED	RO	Part number, bits[11:8]. Set by the configuration inputs part_number[11:8].

9.8.39 css600_apbrom_gpr_64bit Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

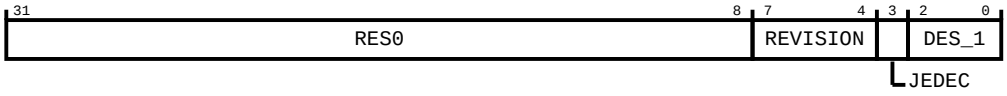
Reset value

0x000000--

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-187: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-193: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	**IMPLEMENTATION DEFINED**	RO	Revision. Set by the configuration inputs revision[3:0].
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	**IMPLEMENTATION DEFINED**	RO	JEP106 identification code, bits[6:4]. Set by the configuration inputs jep106_id[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.8.40 css600_apbrom_gpr_64bit Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFEC

Type

RO

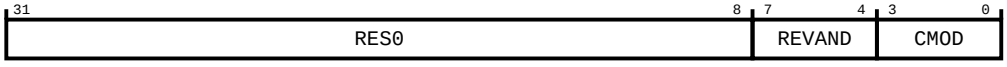
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-188: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-194: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.

Bits	Name	Reset	Type	Description
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.8.41 css600_apbrom_gpr_64bit Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

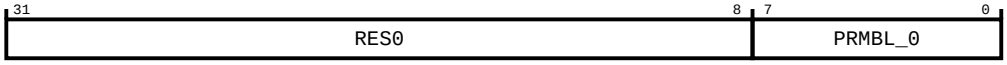
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-189: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-195: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.8.42 css600_apbrom_gpr_64bit Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF4`

Type

RO

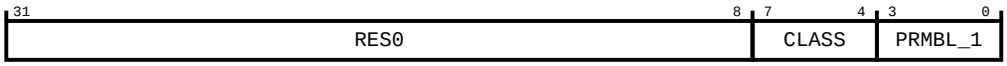
Reset value

`0x00000090`

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-190: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-196: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	`0b1001`	RO	Component class **`0b1001`** CoreSight component
3:0	PRMBL_1	`0b0000`	RO	Preamble. Returns `0x0`.

9.8.43 css600_apbrom_gpr_64bit Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

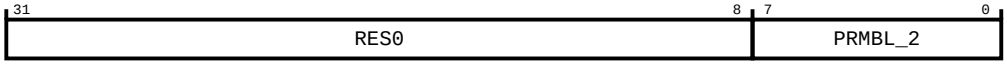
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-191: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-197: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.8.44 css600_apbrom_gpr_64bit Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

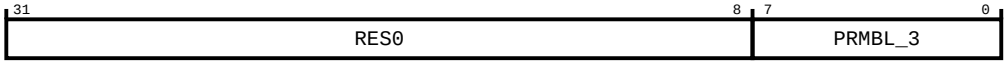
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-192: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-198: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.9 css600_apv1adapter register summary

This section describes the css600_apv1adapter_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary tables

Table 9-199: Implementation defined registers

Offset	Name	Type	Reset	Width	Description
0xd00	Downstream reg0	implementation defined	0x--------	32-bit	Accesses APv1 register at 0x-----d00
0xd04	Downstream reg1	implementation defined	0x--------	32-bit	Accesses APv1 register at 0x-----d04
0xd08	Downstream reg2	implementation defined	0x--------	32-bit	Accesses APv1 register at 0x-----d08
0xd0a	Downstream reg3	implementation defined	0x--------	32-bit	Accesses APv1 register at 0x-----d0a
...	...	...	...	32-bit	...
0xdfc	Downstream reg63	implementation defined	0x--------	32-bit	Accesses APv1 register at 0x-----dfc

Table 9-200: css600_apv1adapter_registers register summary

Offset	Name	Type	Reset	Width	Description
0xefc	ITSTATUS	RO	0x00000000	32-bit	Integration Test Status register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x00000003	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x00000000	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x47700A47	32-bit	Device Architecture Register
0xfcc	DEVTYPE	RO	0x00000000	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5

Offset	Name	Type	Reset	Width	Description
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000E5	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000001B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.9.1 css600_apv1adapter Integration Test Status register, ITSTATUS

This register indicates the Integration Test DP Abort status.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEFC

Type

RO

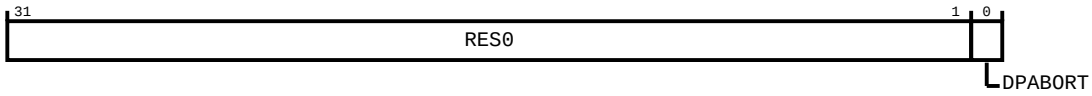
Reset value

0x00000000

Bit descriptions

The following figure shows the ITSTATUS register bit assignments.

Figure 9-193: Bit assignment diagram for the ITSTATUS register



The following table shows the ITSTATUS register bit descriptions.

Table 9-201: ITSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
0	DPABORT	0b0	RO	When in Integration testing mode (ITCTRL.IME=0b1): Behaves as a sticky bit and latches to 1 on a rising edge of dp_abort. Cleared on a read from this register. If dp_abort rises in the same cycle as a read of the ITSTATUS register is received, the read takes priority and the register is cleared. When in normal functional operation mode (ITCTRL.IME=0b0): Read as 0, writes ignored.

9.9.2 css600_apv1adapter Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

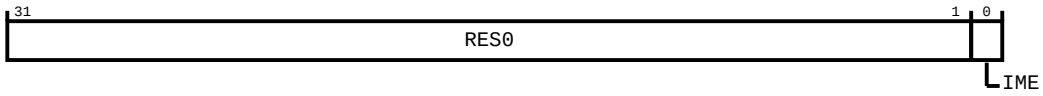
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-194: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-202: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.9.3 css600_apv1adapter Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

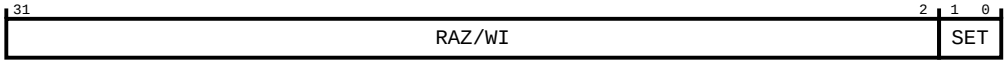
Reset value

0x00000003

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-195: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-203: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RAZ/WI**	0x0	RO	**RAZ/WI.**
1:0	SET	0b11	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.9.4 css600_apv1adapter Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFA4`

Type

RW

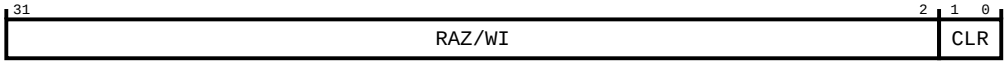
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-196: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-204: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RAZ/WI**	`0x0`	RO	**RAZ/WI.**
1:0	CLR	`0b00`	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.9.5 css600_apv1adapter Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

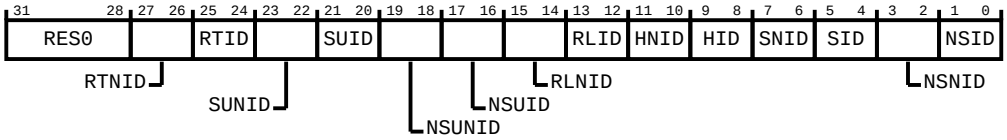
Reset value

0x00000000

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-197: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-205: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	0b00	RO	Root non-invasive debug. **0b00** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.
25:24	RTID	0b00	RO	Root invasive debug. **0b00** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	0b00	RO	Secure non-invasive debug. **0b00** Debug level is not supported.
5:4	SID	0b00	RO	Secure invasive debug. **0b00** Debug level is not supported.
3:2	NSNID	0b00	RO	Non-secure non-invasive debug. **0b00** Debug level is not supported.
1:0	NSID	0b00	RO	Non-secure invasive debug. **0b00** Debug level is not supported.

9.9.6 css600_apv1adapter Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFBC

Type

RO

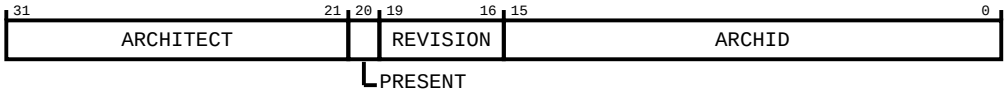
Reset value

0x47700A47

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-198: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-206: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x23B	RO	Defines the architect of the component **0x23B** Arm
20	PRESENT	0b1	RO	Indicates the presence of this register **0b1** DEVARCH is present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0A47	RO	Architecture ID. Returns a value that identifies the architecture of the component. **0x0A47** Unknown Access Port v2 architecture

9.9.7 css600_apv1adapter Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

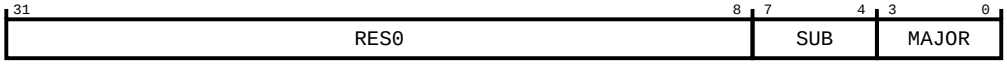
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-199: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-207: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	0b0000	RO	Minor classification. Returns 0x0, Other/undefined.
3:0	MAJOR	0b0000	RO	Major classification. Returns 0x0, Miscellaneous.

9.9.8 css600_apv1adapter Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

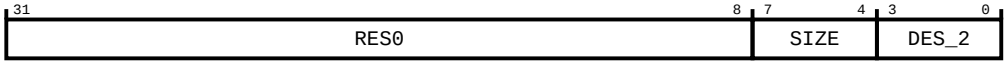
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-200: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-208: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.9.9 css600_apv1adapter Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

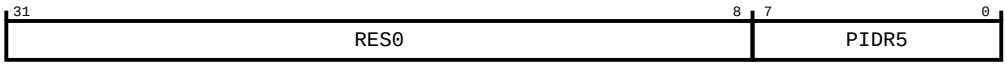
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-201: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-209: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.9.10 css600_apv1adapter Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

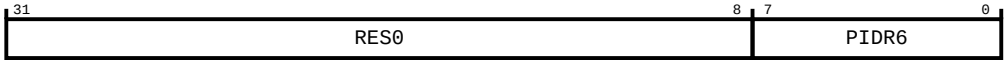
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-202: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-210: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.9.11 css600_apv1adapter Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFDC`

Type

RO

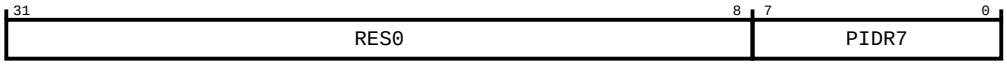
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-203: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-211: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.9.12 css600_apv1adapter Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

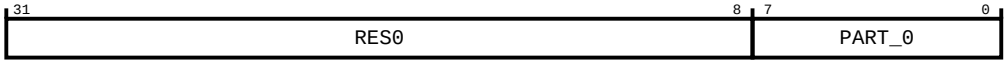
Reset value

0x000000E5

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-204: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-212: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xE5	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.9.13 css600_apv1adapter Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

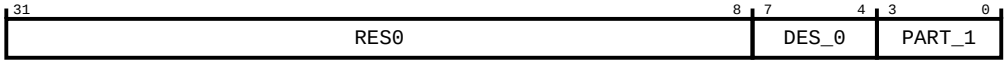
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-205: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-213: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.9.14 css600_apv1adapter Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

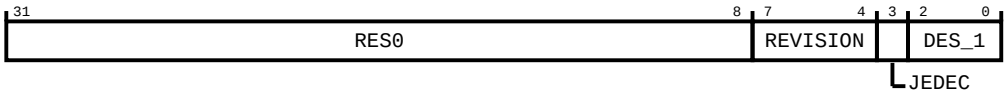
Reset value

0x0000001B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-206: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-214: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b0001	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.9.15 css600_apv1adapter Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

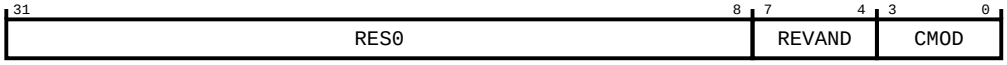
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-207: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-215: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	`0b0000`	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is `0x0`.
3:0	CMOD	`0b0000`	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is `0x0`.

9.9.16 css600_apv1adapter Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

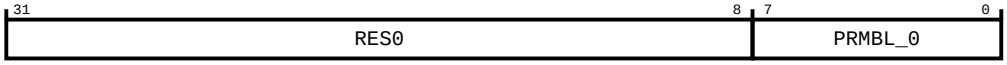
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-208: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-216: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.9.17 css600_apv1adapter Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

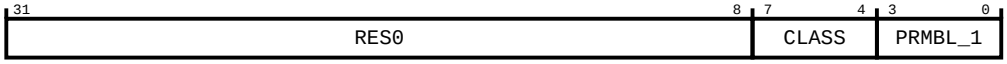
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-209: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-217: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.9.18 css600_apv1adapter Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

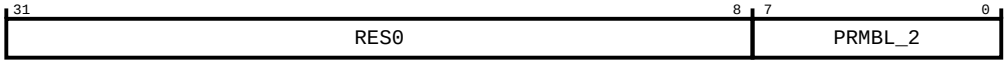
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-210: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-218: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.9.19 css600_apv1adapter Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

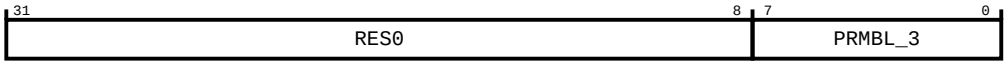
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-211: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-219: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.10 css600_atbfunnel_prog register summary

This section describes the css600_atbfunnel_prog_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-220: css600_atbfunnel_prog_registers register summary

Offset	Name	Type	Reset	Width	Description
0x000	FUNNELCONTROL	RW	0x00000300	32-bit	Funnel Control register
0x004	PRIORITYCONTROL	RW	0x00000000	32-bit	Priority Control register
0xeec	ITATBDATA0	RW	0x00000000	32-bit	Integration test data register
0xef0	ITATBCTR3	RW	0x00000000	32-bit	Integration test control register 3
0xef4	ITATBCTR2	RW	0x00000000	32-bit	Integration test control register 2
0xef8	ITATBCTR1	RW	0x00000000	32-bit	Integration test control register 1
0xefc	ITATBCTR0	RW	0x00000000	32-bit	Integration test control register 0
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x0000000F	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfa8	DEVAFF0	RO	0x00000000	32-bit	Device Affinity register 0
0xfac	DEVAFF1	RO	0x00000000	32-bit	Device Affinity register 1
0xfb8	AUTHSTATUS	RO	0x00000000	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x00000000	32-bit	Device Architecture Register
0xfc0	DEVID2	RO	0x00000000	32-bit	Device Configuration Register 2
0xfc4	DEVID1	RO	0x00000000	32-bit	Device Configuration Register 1
0xfc8	DEVID	RO	0x0000003-	32-bit	Device Configuration Register
0xfcc	DEVTYPE	RO	0x00000012	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000EB	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000005B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.10.1 css600_atbfunnel_prog Funnel Control register, FUNNELCONTROL

The FUNNELCONTROL register enables each of the trace sources and controls the hold time for switching between them.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x000

Type

RW

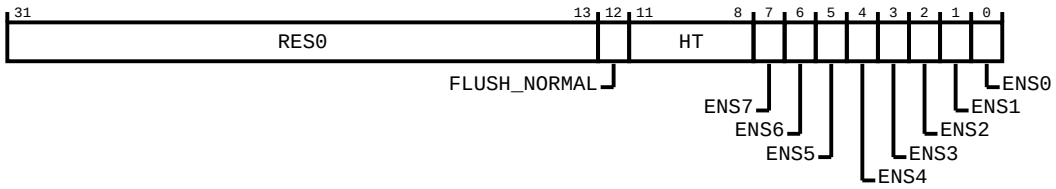
Reset value

0x00000300

Bit descriptions

The following figure shows the FUNNELCONTROL register bit assignments.

Figure 9-212: Bit assignment diagram for the FUNNELCONTROL register



The following table shows the FUNNELCONTROL register bit descriptions.

Table 9-221: FUNNELCONTROL bit descriptions

Bits	Name	Reset	Type	Description
31:13	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
12	FLUSH_NORMAL	0b0	RW	This bit, when clear, allows receiver ports that are already flushed to receive further data even if there are other ports that have not completed flush. If set, a port that has completed flush is not be allowed to receive further data until all ports have completed flush.

Bits	Name	Reset	Type	Description
11:8	HT	0b0011	RW	Hold time. Value sets the minimum hold time before switching trace sources (funnel inputs) based on the ID. Value used is programmed value + 1. **0b0000** 1 transaction hold time **0b0001** 2 transactions hold time **0b0010** 3 transactions hold time **0b0011** 4 transactions hold time **0b0100** 5 transactions hold time **0b0101** 6 transactions hold time **0b0110** 7 transactions hold time **0b0111** 8 transactions hold time **0b1000** 9 transactions hold time **0b1001** 10 transactions hold time **0b1010** 11 transactions hold time **0b1011** 12 transactions hold time **0b1100** 13 transactions hold time **0b1101** 14 transactions hold time **0b1110** 15 transactions hold time **0b1111** Reserved
7	ENS7	0b0	RW	Enable receiver interface 7. Field is **RES0** if receiver interface 7 is not implemented. **0b0** Receiver interface disabled **0b1** Receiver interface enabled

Bits	Name	Reset	Type	Description
6	ENS6	0b0	RW	Enable receiver interface 6. Field is **RES0** if receiver interface 6 is not implemented. **0b0** Receiver interface disabled **0b1** Receiver interface enabled
5	ENS5	0b0	RW	Enable receiver interface 5. Field is **RES0** if receiver interface 5 is not implemented. **0b0** Receiver interface disabled **0b1** Receiver interface enabled
4	ENS4	0b0	RW	Enable receiver interface 4. Field is **RES0** if receiver interface 4 is not implemented. **0b0** Receiver interface disabled **0b1** Receiver interface enabled
3	ENS3	0b0	RW	Enable receiver interface 3. Field is **RES0** if receiver interface 3 is not implemented. **0b0** Receiver interface disabled **0b1** Receiver interface enabled
2	ENS2	0b0	RW	Enable receiver interface 2. Field is **RES0** if receiver interface 2 is not implemented. **0b0** Receiver interface disabled **0b1** Receiver interface enabled
1	ENS1	0b0	RW	Enable receiver interface 1. **0b0** Receiver interface disabled **0b1** Receiver interface enabled
0	ENS0	0b0	RW	Enable receiver interface 0. **0b0** Receiver interface disabled **0b1** Receiver interface enabled

9.10.2 css600_atbfunnel_prog Priority Control register, PRIORITYCONTROL

The PRIORITYCONTROL register sets the priority of each port (receiver interface) of the funnel. The programming software requires that the ports are all disabled before the priority control register contents are changed. Changing the port priorities in real time is not supported.

If the priority control register is written when one or more of the ports are enabled, then the write is silently rejected and the value in the priority control register remains unchanged. The lower the priority value, the higher is its priority when selecting the next port to be serviced. If two or more ports have the same priority value, then the lowest numbered port is serviced first.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x004

Type

RW

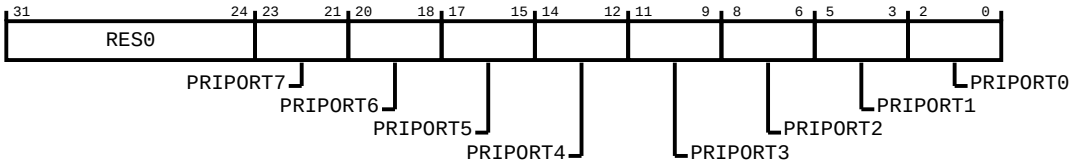
Reset value

0x00000000

Bit descriptions

The following figure shows the PRIORITYCONTROL register bit assignments.

Figure 9-213: Bit assignment diagram for the PRIORITYCONTROL register



The following table shows the PRIORITYCONTROL register bit descriptions.

Table 9-222: PRIORITYCONTROL bit descriptions

Bits	Name	Reset	Type	Description
31:24	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
23:21	PRIPORT7	0b000	RW	Priority value for port 7. Field is **RES0** if port 7 is not implemented.
20:18	PRIPORT6	0b000	RW	Priority value for port 6. Field is **RES0** if port 6 is not implemented.
17:15	PRIPORT5	0b000	RW	Priority value for port 5. Field is **RES0** if port 5 is not implemented.
14:12	PRIPORT4	0b000	RW	Priority value for port 4. Field is **RES0** if port 4 is not implemented.

Bits	Name	Reset	Type	Description
11:9	PRIPORT3	0b000	RW	Priority value for port 3. Field is **RES0** if port 3 is not implemented.
8:6	PRIPORT2	0b000	RW	Priority value for port 2. Field is **RES0** if port 2 is not implemented.
5:3	PRIPORT1	0b000	RW	Priority value for port 1.
2:0	PRIPORT0	0b000	RW	Priority value for port 0.

9.10.3 css600_atbfunnel_prog Integration test data register, ITATBDATA0

The ITATBDATA0 register allows observability and controllability of the ATDATA buses into and out of the funnel. For receiver signals coming into the funnel, the register views the ports that are selected through the funnel control register. Only one port must be selected for integration test.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEEC

Type

RW

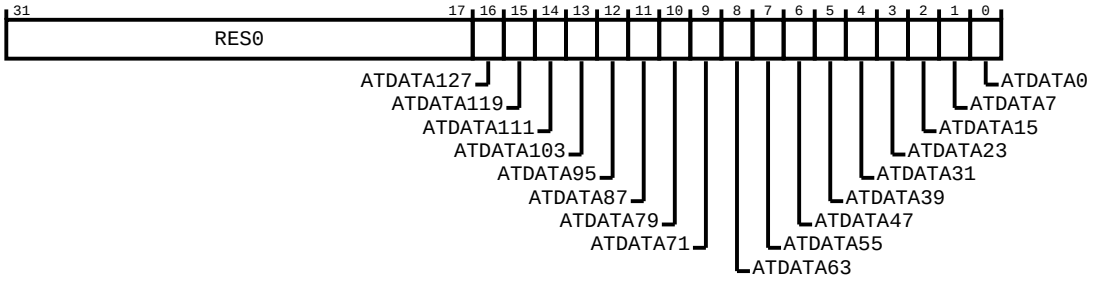
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBDATA0 register bit assignments.

Figure 9-214: Bit assignment diagram for the ITATBDATA0 register



The following table shows the ITATBDATA0 register bit descriptions.

Table 9-223: ITATBDATA0 bit descriptions

Bits	Name	Reset	Type	Description
31:17	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
16	ATDATA127	0b0	RW	Reads atdata_rx[127] and writes atdata_tx[127]. **RES0** if ATB_DATA_WIDTH < 128. **0b0** On reads, the value of atdata_rx[127] is 0. On writes, sets atdata_tx[127] to 0 **0b1** On reads, the value of atdata_rx[127] is 1. On writes, sets atdata_tx[127] to 1
15	ATDATA119	0b0	RW	Reads atdata_rx[119] and writes atdata_tx[119]. **RES0** if ATB_DATA_WIDTH < 128. **0b0** On reads, the value of atdata_rx[119] is 0. On writes, sets atdata_tx[119] to 0 **0b1** On reads, the value of atdata_rx[119] is 1. On writes, sets atdata_tx[119] to 1
14	ATDATA111	0b0	RW	Reads atdata_rx[111] and writes atdata_tx[111]. **RES0** if ATB_DATA_WIDTH < 128. **0b0** On reads, the value of atdata_rx[111] is 0. On writes, sets atdata_tx[111] to 0 **0b1** On reads, the value of atdata_rx[111] is 1. On writes, sets atdata_tx[111] to 1
13	ATDATA103	0b0	RW	Reads atdata_rx[103] and writes atdata_tx[103]. **RES0** if ATB_DATA_WIDTH < 128. **0b0** On reads, the value of atdata_rx[103] is 0. On writes, sets atdata_tx[103] to 0 **0b1** On reads, the value of atdata_rx[103] is 1. On writes, sets atdata_tx[103] to 1
12	ATDATA95	0b0	RW	Reads atdata_rx[95] and writes atdata_tx[95]. **RES0** if ATB_DATA_WIDTH < 128. **0b0** On reads, the value of atdata_rx[95] is 0. On writes, sets atdata_tx[95] to 0 **0b1** On reads, the value of atdata_rx[95] is 1. On writes, sets atdata_tx[95] to 1
11	ATDATA87	0b0	RW	Reads atdata_rx[87] and writes atdata_tx[87]. **RES0** if ATB_DATA_WIDTH < 128. **0b0** On reads, the value of atdata_rx[87] is 0. On writes, sets atdata_tx[87] to 0 **0b1** On reads, the value of atdata_rx[87] is 1. On writes, sets atdata_tx[87] to 1
10	ATDATA79	0b0	RW	Reads atdata_rx[79] and writes atdata_tx[79]. **RES0** if ATB_DATA_WIDTH < 128. **0b0** On reads, the value of atdata_rx[79] is 0. On writes, sets atdata_tx[79] to 0 **0b1** On reads, the value of atdata_rx[79] is 1. On writes, sets atdata_tx[79] to 1
9	ATDATA71	0b0	RW	Reads atdata_rx[71] and writes atdata_tx[71]. **RES0** if ATB_DATA_WIDTH < 128. **0b0** On reads, the value of atdata_rx[71] is 0. On writes, sets atdata_tx[71] to 0 **0b1** On reads, the value of atdata_rx[71] is 1. On writes, sets atdata_tx[71] to 1

Bits	Name	Reset	Type	Description
8	ATDATA63	0b0	RW	Reads atdata_rx[63] and writes atdata_tx[63]. **RES0** if ATB_DATA_WIDTH < 64. **0b0** On reads, the value of atdata_rx[63] is 0. On writes, sets atdata_tx[63] to 0 **0b1** On reads, the value of atdata_rx[63] is 1. On writes, sets atdata_tx[63] to 1
7	ATDATA55	0b0	RW	Reads atdata_rx[55] and writes atdata_tx[55]. **RES0** if ATB_DATA_WIDTH < 64. **0b0** On reads, the value of atdata_rx[55] is 0. On writes, sets atdata_tx[55] to 0 **0b1** On reads, the value of atdata_rx[55] is 1. On writes, sets atdata_tx[55] to 1
6	ATDATA47	0b0	RW	Reads atdata_rx[47] and writes atdata_tx[47]. **RES0** if ATB_DATA_WIDTH < 64. **0b0** On reads, the value of atdata_rx[47] is 0. On writes, sets atdata_tx[47] to 0 **0b1** On reads, the value of atdata_rx[47] is 1. On writes, sets atdata_tx[47] to 1
5	ATDATA39	0b0	RW	Reads atdata_rx[39] and writes atdata_tx[39]. **RES0** if ATB_DATA_WIDTH < 64. **0b0** On reads, the value of atdata_rx[39] is 0. On writes, sets atdata_tx[39] to 0 **0b1** On reads, the value of atdata_rx[39] is 1. On writes, sets atdata_tx[39] to 1
4	ATDATA31	0b0	RW	Reads atdata_rx[31] and writes atdata_tx[31]. **RES0** if ATB_DATA_WIDTH < 32. **0b0** On reads, the value of atdata_rx[31] is 0. On writes, sets atdata_tx[31] to 0 **0b1** On reads, the value of atdata_rx[31] is 1. On writes, sets atdata_tx[31] to 1
3	ATDATA23	0b0	RW	Reads atdata_rx[23] and writes atdata_tx[23]. **RES0** if ATB_DATA_WIDTH < 32. **0b0** On reads, the value of atdata_rx[23] is 0. On writes, sets atdata_tx[23] to 0 **0b1** On reads, the value of atdata_rx[23] is 1. On writes, sets atdata_tx[23] to 1
2	ATDATA15	0b0	RW	Reads atdata_rx[15] and writes atdata_tx[15]. **RES0** if ATB_DATA_WIDTH < 16. **0b0** On reads, the value of atdata_rx[15] is 0. On writes, sets atdata_tx[15] to 0 **0b1** On reads, the value of atdata_rx[15] is 1. On writes, sets atdata_tx[15] to 1
1	ATDATA7	0b0	RW	Reads atdata_rx[7] and writes atdata_tx[7]. **0b0** On reads, the value of atdata_rx[7] is 0. On writes, sets atdata_tx[7] to 0 **0b1** On reads, the value of atdata_rx[7] is 1. On writes, sets atdata_tx[7] to 1

Bits	Name	Reset	Type	Description
0	ATDATA0	0b0	RW	Reads atdata_rx[0] and writes atdata_tx[0]. **0b0** On reads, the value of atdata_rx[0] is 0. On writes, sets atdata_tx[0] to 0 **0b1** On reads, the value of atdata_rx[0] is 1. On writes, sets atdata_tx[0] to 1

9.10.4 css600_atbfunnel_prog Integration test control register 3, ITATBCTR3

The ITATBCTR3 register enables you to observe and control the SYNCREQ signals into and out of the funnel. Only one receiver interface must be selected for integration test. The syncreq receiver on the transmitter interface has a latching function to capture a pulse arriving on that input.

The arrival of a pulse sets the latch so that the value can be read. Reading the register clears the latch. Reading a 1 indicates that a syncreq_tx pulse arrived since the last read. Reading a 0 indicates that no syncreq_tx pulse has arrived. Writing a 1 to the register causes a syncreq_rx pulse to be generated to the upstream component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF0

Type

RW

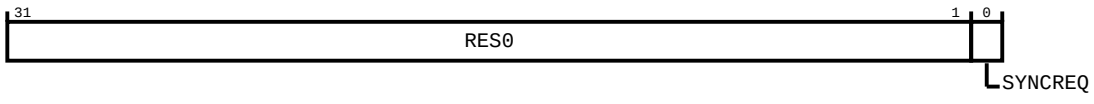
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR3 register bit assignments.

Figure 9-215: Bit assignment diagram for the ITATBCTR3 register



The following table shows the ITATBCTR3 register bit descriptions.

Table 9-224: ITATBCTR3 bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	SYNCREQ	0b0	RW	Reads and controls the SYNCREQ signals into, and out of, the funnel. Reading clears the latch. **0b0** On reads: no syncreq_tx pulse has arrived. On writes: no effect. **0b1** On reads: a syncreq_tx pulse arrived since the last read. On writes: generates a syncreq_rx pulse to the upstream component.

9.10.5 css600_atbfunnel_prog Integration test control register 2, ITATBCTR2

The ITATBCTR2 register enables you to observe and control the afvalid and atready signals into and out of the funnel. For receiver signals coming into the funnel, the register views the ports that are selected through the FUNNELCONTROL register. Only one port must be selected for integration test.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF4

Type

RW

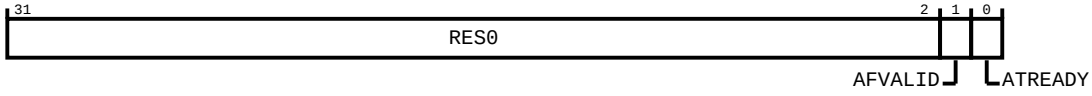
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR2 register bit assignments.

Figure 9-216: Bit assignment diagram for the ITATBCTR2 register



The following table shows the ITATBCTR2 register bit descriptions.

Table 9-225: ITATBCTR2 bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	AFVALID	0b0	RW	Reads and controls the afvalid signals into, and out of, the funnel: **0b0** On reads: afvalid_tx is LOW. On writes: sets afvalid_rx LOW. **0b1** On reads: afvalid_tx is HIGH. On writes: sets afvalid_rx HIGH.
0	ATREADY	0b0	RW	Reads and controls the atready signal into, and out of, the funnel: **0b0** On reads: atready_tx is LOW. On writes: sets atready_rx LOW. **0b1** On reads: atready_tx is HIGH. On writes: sets atready_rx HIGH.

9.10.6 css600_atbfunnel_prog Integration test control register 1, ITATBCTR1

The ITATBCTR1 register enables you to observe and control the ATID buses into and out of the funnel. For receiver signals coming into the funnel, the register views the ports that are selected through the FUNNELCONTROL register. Only one port must be selected for integration test.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF8

Type

RW

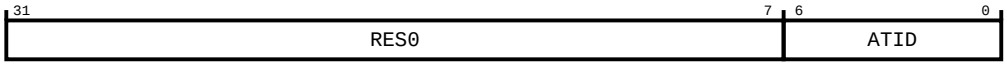
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR1 register bit assignments.

Figure 9-217: Bit assignment diagram for the ITATBCTR1 register



The following table shows the ITATBCTR1 register bit descriptions.

Table 9-226: ITATBCTR1 bit descriptions

Bits	Name	Reset	Type	Description
31:7	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
6:0	ATID	0x0	RW	When read returns the value on atid_rx, when written drives the value on atid_tx

9.10.7 css600_atbfunnel_prog Integration test control register 0, ITATBCTR0

The ITATBCTR0 register enables you to observe and control the ATBYTES buses, and the AFREADY and ATVALID signals into and out of the funnel. For receiver signals coming into the funnel, the register views the ports that are selected through the FUNNELCONTROL register.

Only one port must be selected for integration test.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEFC

Type

RW

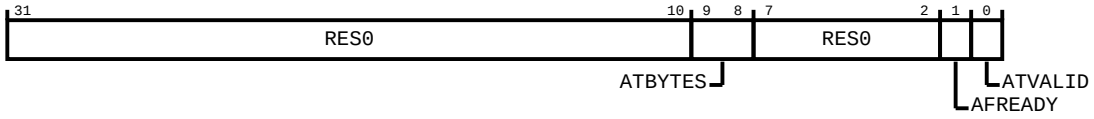
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR0 register bit assignments.

Figure 9-218: Bit assignment diagram for the ITATBCTR0 register



The following table shows the ITATBCTR0 register bit descriptions.

Table 9-227: ITATBCTR0 bit descriptions

Bits	Name	Reset	Type	Description
31:10	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
9:8	ATBYTES	0b00	RW	Reads the value on atbytes_rx[1:0] and writes the values on atbytes_tx[1:0]
7:2	**RES0**	0b000000	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
1	AFREADY	0b0	RW	Reads and controls the afready signals into, and out of, the funnel: **0b0** On reads: afready_rx is LOW. On writes: sets afready_tx LOW. **0b1** On reads: afready_rx is HIGH. On writes: sets afready_tx HIGH.
0	ATVALID	0b0	RW	Reads and controls the atvalid signals into, and out of, the funnel: **0b0** On reads: atvalid_rx is LOW. On writes: sets atvalid_tx LOW. **0b1** On reads: atvalid_rx is HIGH. On writes: sets atvalid_tx HIGH.

9.10.8 css600_atbfunnel_prog Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

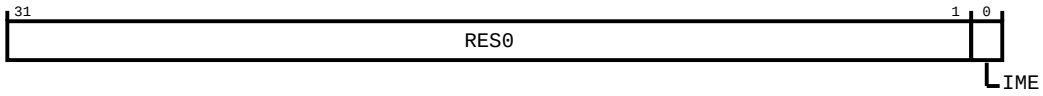
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-219: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-228: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.10.9 css600_atbfunnel_prog Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

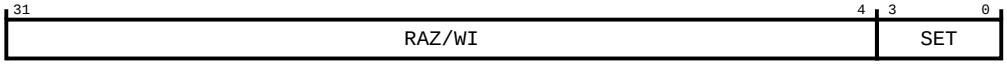
Reset value

0x0000000F

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-220: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-229: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	0x0	RO	**RAZ/WI**.
3:0	SET	0b1111	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.10.10 css600_atbfunnel_prog Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFA4`

Type

RW

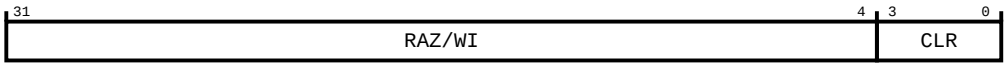
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-221: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-230: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	`0x0`	RO	**RAZ/WI.**
3:0	CLR	`0b0000`	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.10.11 css600_atbfunnel_prog Device Affinity register 0, DEVAFF0

Enables a debugger to determine if two components have an affinity with each other.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA8

Type

RO

Reset value

0x00000000

Bit descriptions

The following figure shows the DEVAFF0 register bit assignments.

Figure 9-222: Bit assignment diagram for the DEVAFF0 register

The following table shows the DEVAFF0 register bit descriptions.

Table 9-231: DEVAFF0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	DEVAFF0	0x0	RO	This field is **RAZ**.

9.10.12 css600_atbfunnel_prog Device Affinity register 1, DEVAFF1

Enables a debugger to determine if two components have an affinity with each other.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFAC

Type

RO

Reset value

0x00000000

Bit descriptions

The following figure shows the DEVAFF1 register bit assignments.

Figure 9-223: Bit assignment diagram for the DEVAFF1 register

The following table shows the DEVAFF1 register bit descriptions.

Table 9-232: DEVAFF1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	DEVAFF1	0x0	RO	This field is **RAZ**.

9.10.13 css600_atbfunnel_prog Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

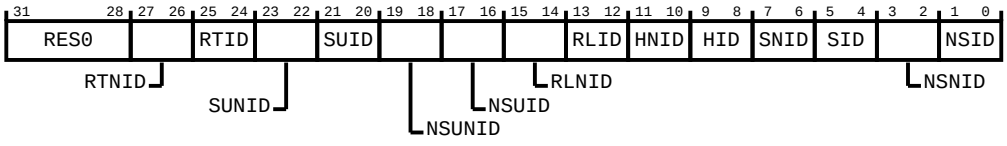
Reset value

0x00000000

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-224: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-233: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	0b00	RO	Root non-invasive debug. **0b00** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.
25:24	RTID	0b00	RO	Root invasive debug. **0b00** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	0b00	RO	Secure non-invasive debug. **0b00** Debug level is not supported.
5:4	SID	0b00	RO	Secure invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
3:2	NSNID	0b00	RO	Non-secure non-invasive debug. **0b00** Debug level is not supported.
1:0	NSID	0b00	RO	Non-secure invasive debug. **0b00** Debug level is not supported.

9.10.14 css600_atbfunnel_prog Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFBC

Type

RO

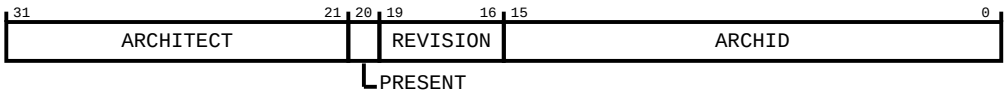
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-225: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-234: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x000	RO	Defines the architect of the component
20	PRESENT	0b0	RO	Indicates the presence of this register **0b0** DEVARCH is not present

Bits	Name	Reset	Type	Description
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0000	RO	Architecture ID. Returns a value that identifies the architecture of the component.

9.10.15 css600_atbfunnel_prog Device Configuration Register 2, DEVID2

Contains an **IMPLEMENTATION DEFINED** value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC0

Type

RO

Reset value

0x00000000

Bit descriptions

The following figure shows the DEVID2 register bit assignments.

Figure 9-226: Bit assignment diagram for the DEVID2 register

The following table shows the DEVID2 register bit descriptions.

Table 9-235: DEVID2 bit descriptions

Bits	Name	Reset	Type	Description
31:0	DEVID2	0x0	RO	This field is **RAZ**.

9.10.16 css600_atbfunnel_prog Device Configuration Register 1, DEVID1

Contains an **IMPLEMENTATION DEFINED** value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC4

Type

RO

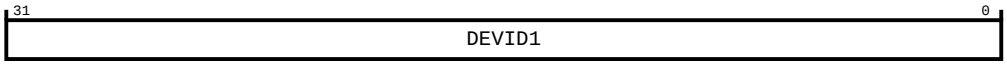
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVID1 register bit assignments.

Figure 9-227: Bit assignment diagram for the DEVID1 register



The following table shows the DEVID1 register bit descriptions.

Table 9-236: DEVID1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	DEVID1	0x0	RO	This field is **RAZ**.

9.10.17 css600_atbfunnel_prog Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC8

Type

RO

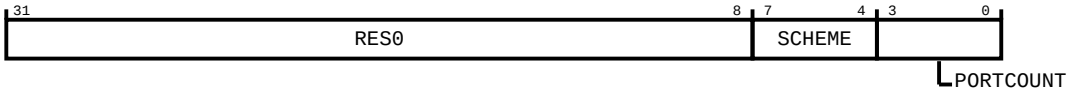
Reset value

0x0000003-

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-228: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-237: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SCHEME	0b0011	RO	Indicates priority scheme implemented. Input priority is controlled by the PRIORITYCONTROL register.
3:0	PORTCOUNT	**IMPLEMENTATION DEFINED**	RO	Indicates the number of input ports connected

9.10.18 css600_atbfunnel_prog Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

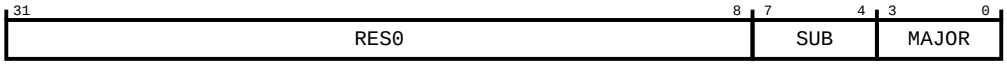
Reset value

0x00000012

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-229: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-238: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	0b0001	RO	Minor classification. Returns 0x1, indicating this component is a Funnel/Router.
3:0	MAJOR	0b0010	RO	Major classification. Returns 0x2, indicating this component is a Trace Link.

9.10.19 css600_atbfunnel_prog Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

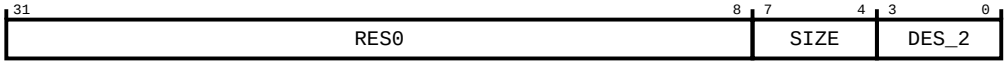
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-230: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-239: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.10.20 css600_atbfunnel_prog Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD4`

Type

RO

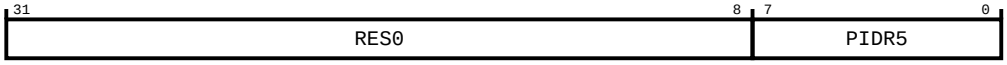
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-231: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-240: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.10.21 css600_atbfunnel_prog Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

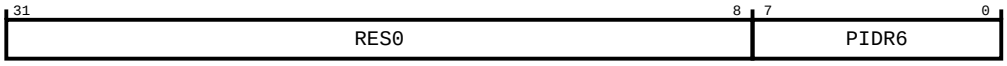
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-232: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-241: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.10.22 css600_atbfunnel_prog Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

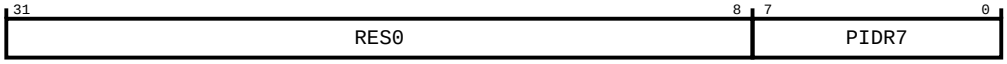
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-233: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-242: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.10.23 css600_atbfunnel_prog Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

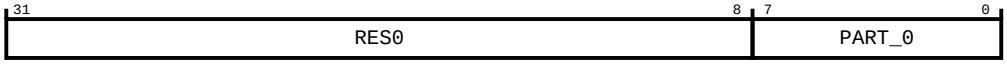
Reset value

0x000000EB

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-234: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-243: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:0	PART_0	0xEB	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.10.24 css600_atbfunnel_prog Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

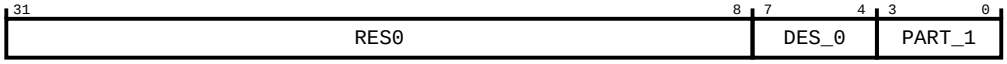
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-235: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-244: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.10.25 css600_atbfunnel_prog Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFE8`

Type

RO

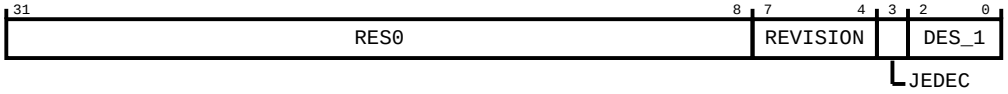
Reset value

`0x0000005B`

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-236: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-245: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	`0b0101`	RO	Revision. It is an incremental value starting at `0x0` for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	`0b1`	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	`0b011`	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.10.26 css600_atbfunnel_prog Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFEC

Type

RO

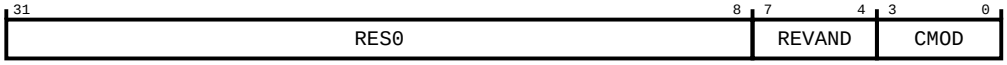
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-237: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-246: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.10.27 css600_atbfunnel_prog Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

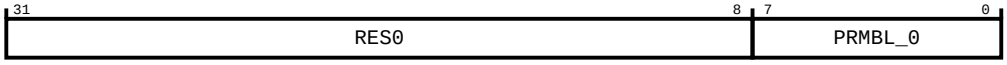
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-238: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-247: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.10.28 css600_atbfunnel_prog Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

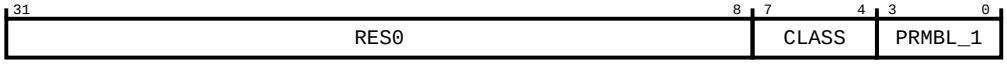
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-239: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-248: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.10.29 css600_atbfunnel_prog Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

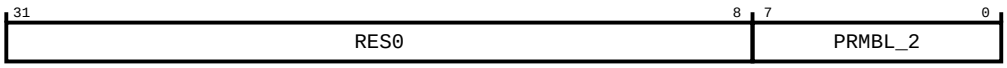
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-240: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-249: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.10.30 css600_atbfunnel_prog Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

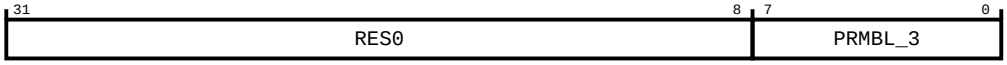
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-241: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-250: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.11 css600_atbreplicator_prog register summary

This section describes the css600_atbreplicator_prog_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-251: css600_atbreplicator_prog_registers register summary

Offset	Name	Type	Reset	Width	Description
0x000	IDFILT0	RW	0x00000000	32-bit	ID filtering control 0 register
0x004	IDFILT1	RW	0x00000000	32-bit	ID filtering control 1 register
0xef8	ITATBCTRL	RW	0x00000000	32-bit	Integration Test Control register
0xefc	ITATBSTAT	RO	0x00000000	32-bit	Integration Test Status register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x0000000F	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfa8	DEVAFF0	RO	0x00000000	32-bit	Device Affinity register 0
0xfac	DEVAFF1	RO	0x00000000	32-bit	Device Affinity register 1
0xfb8	AUTHSTATUS	RO	0x00000000	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x00000000	32-bit	Device Architecture Register
0xfc0	DEVID2	RO	0x00000000	32-bit	Device Configuration Register 2
0xfc4	DEVID1	RO	0x00000000	32-bit	Device Configuration Register 1
0xfc8	DEVID	RO	0x00000032	32-bit	Device Configuration Register
0xfcc	DEVTYPE	RO	0x00000022	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000EC	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000005B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.11.1 css600_atbreplicator_prog ID filtering control 0 register, IDFILT0

The IDFILT0 register controls ID filtering for transmitter interface 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x000

Type

RW

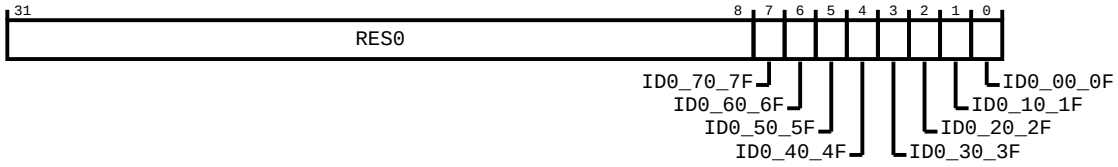
Reset value

0x00000000

Bit descriptions

The following figure shows the IDFILT0 register bit assignments.

Figure 9-242: Bit assignment diagram for the IDFILT0 register



The following table shows the IDFILT0 register bit descriptions.

Table 9-252: IDFILT0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7	ID0_70_7F	0b0	RW	Enable/disable ID filtering for IDs 0x70 to 0x7F: **0b0** Transactions with these IDs are passed on to transmitter interface 0 **0b1** Transactions with these IDs are discarded by the replicator
6	ID0_60_6F	0b0	RW	Enable/disable ID filtering for IDs 0x60 to 0x6F: **0b0** Transactions with these IDs are passed on to transmitter interface 0 **0b1** Transactions with these IDs are discarded by the replicator

Bits	Name	Reset	Type	Description
5	ID0_50_5F	0b0	RW	Enable/disable ID filtering for IDs 0x50 to 0x5F: **0b0** Transactions with these IDs are passed on to transmitter interface 0 **0b1** Transactions with these IDs are discarded by the replicator
4	ID0_40_4F	0b0	RW	Enable/disable ID filtering for IDs 0x40 to 0x4F: **0b0** Transactions with these IDs are passed on to transmitter interface 0 **0b1** Transactions with these IDs are discarded by the replicator
3	ID0_30_3F	0b0	RW	Enable/disable ID filtering for IDs 0x30 to 0x3F: **0b0** Transactions with these IDs are passed on to transmitter interface 0 **0b1** Transactions with these IDs are discarded by the replicator
2	ID0_20_2F	0b0	RW	Enable/disable ID filtering for IDs 0x20 to 0x2F: **0b0** Transactions with these IDs are passed on to transmitter interface 0 **0b1** Transactions with these IDs are discarded by the replicator
1	ID0_10_1F	0b0	RW	Enable/disable ID filtering for IDs 0x10 to 0x1F: **0b0** Transactions with these IDs are passed on to transmitter interface 0 **0b1** Transactions with these IDs are discarded by the replicator
0	ID0_00_0F	0b0	RW	Enable/disable ID filtering for IDs 0x00 to 0x0F: **0b0** Transactions with these IDs are passed on to transmitter interface 0 **0b1** Transactions with these IDs are discarded by the replicator

9.11.2 css600_atbreplicator_prog ID filtering control 1 register, IDFILT1

The IDFILT1 register controls ID filtering for transmitter interface 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x004

Type

RW

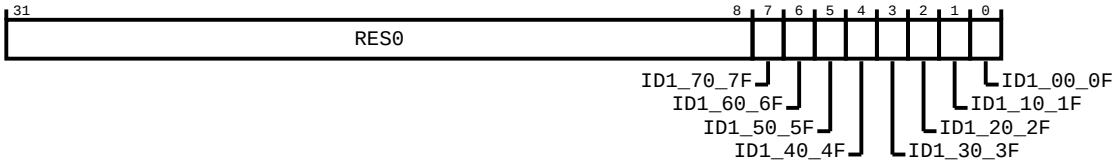
Reset value

0x00000000

Bit descriptions

The following figure shows the IDFILT1 register bit assignments.

Figure 9-243: Bit assignment diagram for the IDFILT1 register



The following table shows the IDFILT1 register bit descriptions.

Table 9-253: IDFILT1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7	ID1_70_7F	0b0	RW	Enable/disable ID filtering for IDs 0x70 to 0x7F: **0b0** Transactions with these IDs are passed on to transmitter interface 1 **0b1** Transactions with these IDs are discarded by the replicator
6	ID1_60_6F	0b0	RW	Enable/disable ID filtering for IDs 0x60 to 0x6F: **0b0** Transactions with these IDs are passed on to transmitter interface 1 **0b1** Transactions with these IDs are discarded by the replicator
5	ID1_50_5F	0b0	RW	Enable/disable ID filtering for IDs 0x50 to 0x5F: **0b0** Transactions with these IDs are passed on to transmitter interface 1 **0b1** Transactions with these IDs are discarded by the replicator
4	ID1_40_4F	0b0	RW	Enable/disable ID filtering for IDs 0x40 to 0x4F: **0b0** Transactions with these IDs are passed on to transmitter interface 1 **0b1** Transactions with these IDs are discarded by the replicator

Bits	Name	Reset	Type	Description
3	ID1_30_3F	0b0	RW	Enable/disable ID filtering for IDs 0x30 to 0x3F: **0b0** Transactions with these IDs are passed on to transmitter interface 1 **0b1** Transactions with these IDs are discarded by the replicator
2	ID1_20_2F	0b0	RW	Enable/disable ID filtering for IDs 0x20 to 0x2F: **0b0** Transactions with these IDs are passed on to transmitter interface 1 **0b1** Transactions with these IDs are discarded by the replicator
1	ID1_10_1F	0b0	RW	Enable/disable ID filtering for IDs 0x10 to 0x1F: **0b0** Transactions with these IDs are passed on to transmitter interface 1 **0b1** Transactions with these IDs are discarded by the replicator
0	ID1_00_0F	0b0	RW	Enable/disable ID filtering for IDs 0x00 to 0x0F: **0b0** Transactions with these IDs are passed on to transmitter interface 1 **0b1** Transactions with these IDs are discarded by the replicator

9.11.3 css600_atbreplicator_prog Integration Test Control register, ITATBCTRL

The ITATBCTRL register controls atready_rx and atvalid_tx.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF8

Type

RW

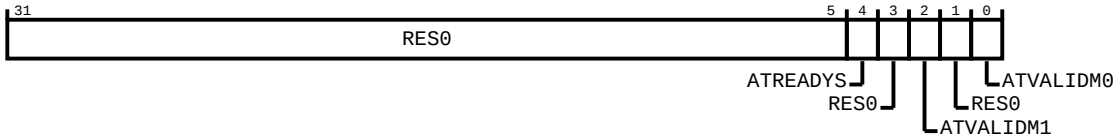
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTRL register bit assignments.

Figure 9-244: Bit assignment diagram for the ITATBCTRL register



The following table shows the ITATBCTRL register bit descriptions.

Table 9-254: ITATBCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:5	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
4	ATREADYS	0b0	RW	On reads: returns the value written to the register. On writes: **0b0** Sets static 0 on atready_rx **0b1** Sets static 1 on atready_rx
3	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
2	ATVALIDM1	0b0	RW	On reads: returns the value written to the register. On writes: **0b0** Sets static 0 on atvalid_tx1 **0b1** Sets static 1 on atvalid_tx1
1	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
0	ATVALIDM0	0b0	RW	On reads: returns the value written to the register. On writes: **0b0** Sets static 0 on atvalid_tx0 **0b1** Sets static 1 on atvalid_tx0

9.11.4 css600_atbreplicator_prog Integration Test Status register, ITATBSTAT

The ITATBSTAT register controls atvalid_rx and atready_tx.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEFC

Type

RO

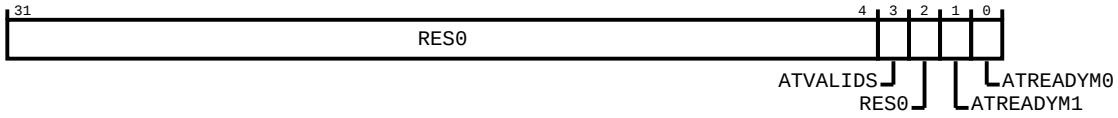
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBSTAT register bit assignments.

Figure 9-245: Bit assignment diagram for the ITATBSTAT register



The following table shows the ITATBSTAT register bit descriptions.

Table 9-255: ITATBSTAT bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
3	ATVALIDS	0b0	RO	Returns the value on atvalid_rx
2	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
1	ATREADYM1	0b0	RO	Returns the value on atready_tx1
0	ATREADYM0	0b0	RO	Returns the value on atready_tx0

9.11.5 css600_atbreplicator_prog Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

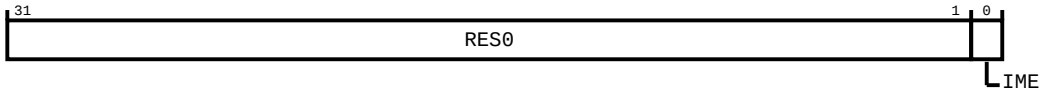
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-246: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-256: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.11.6 css600_atbreplicator_prog Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

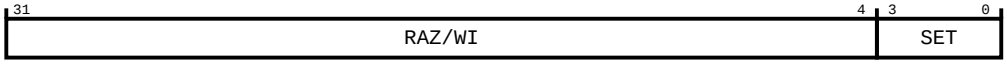
Reset value

0x0000000F

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-247: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-257: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	`0x0`	RO	**RAZ/WI**.
3:0	SET	`0b1111`	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.11.7 css600_atbreplicator_prog Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFA4`

Type

RW

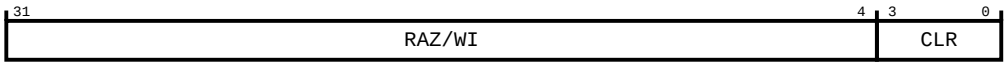
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-248: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-258: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	0x0	RO	**RAZ/WI**.
3:0	CLR	0b0000	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.11.8 css600_atbreplicator_prog Device Affinity register 0, DEVAFF0

Enables a debugger to determine if two components have an affinity with each other.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA8

Type

RO

Reset value

0x00000000

Bit descriptions

The following figure shows the DEVAFF0 register bit assignments.

Figure 9-249: Bit assignment diagram for the DEVAFF0 register

The following table shows the DEVAFF0 register bit descriptions.

Table 9-259: DEVAFF0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	DEVAFF0	0x0	RO	This field is **RAZ**.

9.11.9 css600_atbreplicator_prog Device Affinity register 1, DEVAFF1

Enables a debugger to determine if two components have an affinity with each other.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFAC`

Type

RO

Reset value

`0x00000000`

Bit descriptions

The following figure shows the DEVAFF1 register bit assignments.

Figure 9-250: Bit assignment diagram for the DEVAFF1 register

The following table shows the DEVAFF1 register bit descriptions.

Table 9-260: DEVAFF1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	DEVAFF1	`0x0`	RO	This field is **RAZ**.

9.11.10 css600_atbreplicator_prog Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFB8`

Type

RO

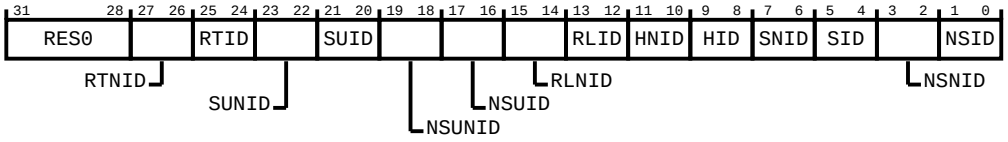
Reset value

0x00000000

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-251: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-261: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	0b00	RO	Root non-invasive debug. **0b00** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.
25:24	RTID	0b00	RO	Root invasive debug. **0b00** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.

Bits	Name	Reset	Type	Description
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	0b00	RO	Secure non-invasive debug. **0b00** Debug level is not supported.
5:4	SID	0b00	RO	Secure invasive debug. **0b00** Debug level is not supported.
3:2	NSNID	0b00	RO	Non-secure non-invasive debug. **0b00** Debug level is not supported.
1:0	NSID	0b00	RO	Non-secure invasive debug. **0b00** Debug level is not supported.

9.11.11 css600_atbreplicator_prog Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFBC

Type

RO

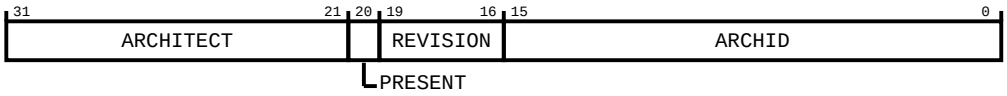
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-252: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-262: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x000	RO	Defines the architect of the component
20	PRESENT	0b0	RO	Indicates the presence of this register **0b0** DEVARCH is not present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0000	RO	Architecture ID. Returns a value that identifies the architecture of the component.

9.11.12 css600_atbreplicator_prog Device Configuration Register 2, DEVID2

Contains an **IMPLEMENTATION DEFINED** value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC0

Type

RO

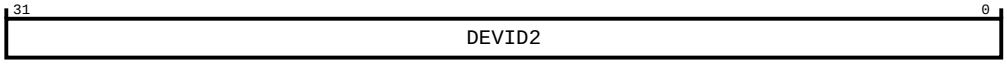
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVID2 register bit assignments.

Figure 9-253: Bit assignment diagram for the DEVID2 register



The following table shows the DEVID2 register bit descriptions.

Table 9-263: DEVID2 bit descriptions

Bits	Name	Reset	Type	Description
31:0	DEVID2	0x0	RO	This field is **RAZ**.

9.11.13 css600_atbreplicator_prog Device Configuration Register 1, DEVID1

Contains an **IMPLEMENTATION DEFINED** value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC4

Type

RO

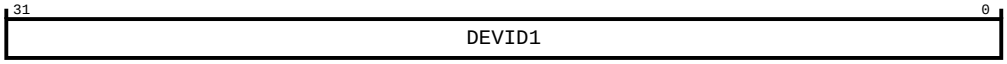
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVID1 register bit assignments.

Figure 9-254: Bit assignment diagram for the DEVID1 register



The following table shows the DEVID1 register bit descriptions.

Table 9-264: DEVID1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	DEVID1	0x0	RO	This field is **RAZ**.

9.11.14 css600_atbreplicator_prog Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFC8`

Type

RO

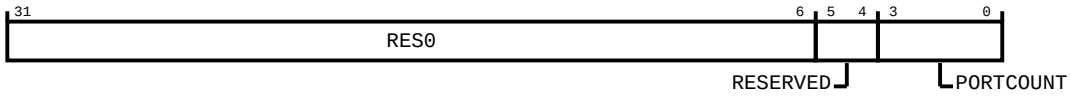
Reset value

`0x00000032`

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-255: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-265: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:6	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
5:4	RESERVED	`0b11`	RO	Reserved. Returns `0x3`. Software must not rely on this value.
3:0	PORTCOUNT	`0b0010`	RO	Indicates the number of transmitter ports implemented

9.11.15 css600_atbreplicator_prog Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFCC`

Type

RO

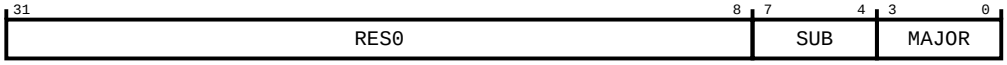
Reset value

`0x00000022`

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-256: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-266: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	`0b0010`	RO	Minor classification. Returns `0x2`, indicates this component is a Filter.
3:0	MAJOR	`0b0010`	RO	Major classification. Returns `0x2`, indicating this component is a Trace Link.

9.11.16 css600_atbreplicator_prog Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD0`

Type

RO

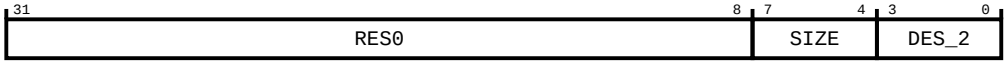
Reset value

`0x00000004`

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-257: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-267: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.11.17 css600_atbreplicator_prog Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

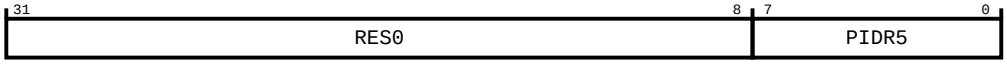
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-258: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-268: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.11.18 css600_atbreplicator_prog Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD8

Type

RO

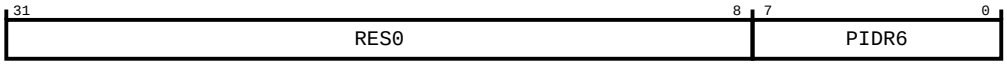
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-259: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-269: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:0	PIDR6	0x0	RO	Reserved.

9.11.19 css600_atbreplicator_prog Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

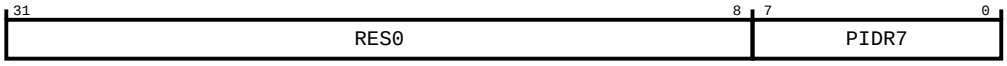
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-260: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-270: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.11.20 css600_atbreplicator_prog Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFE0`

Type

RO

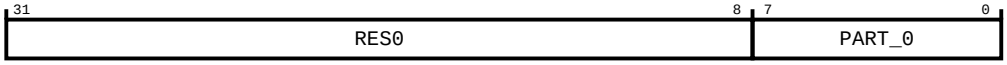
Reset value

`0x000000EC`

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-261: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-271: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xEC	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.11.21 css600_atbreplicator_prog Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFE4`

Type

RO

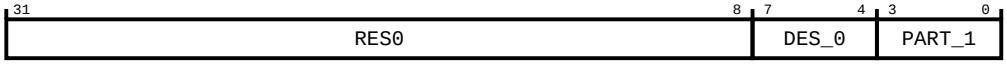
Reset value

`0x000000B9`

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-262: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-272: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.11.22 css600_atbreplicator_prog Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

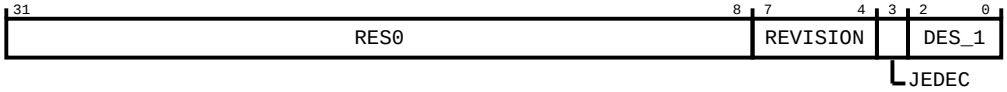
Reset value

0x0000005B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-263: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-273: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b0101	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.11.23 css600_atbreplicator_prog Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFEC

Type

RO

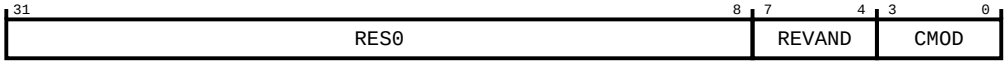
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-264: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-274: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.11.24 css600_atbreplicator_prog Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

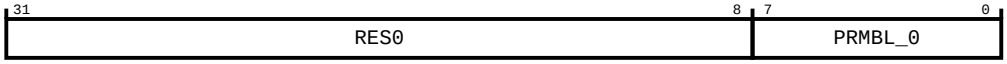
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-265: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-275: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.11.25 css600_atbreplicator_prog Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

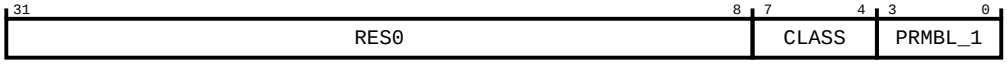
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-266: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-276: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.11.26 css600_atbreplicator_prog Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF8`

Type

RO

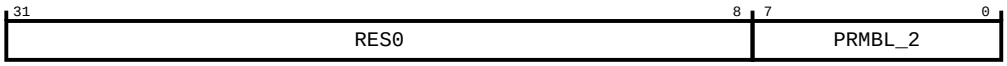
Reset value

`0x00000005`

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-267: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-277: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	`0x5`	RO	Preamble. Returns `0x05`.

9.11.27 css600_atbreplicator_prog Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

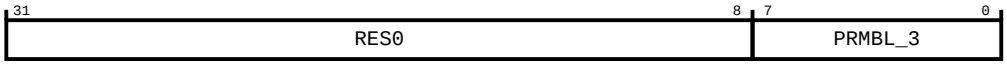
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-268: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-278: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.12 css600_axiap register summary

This section describes the css600_axiap_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-279: css600_axiap_registers register summary

Offset	Name	Type	Reset	Width	Description
0x0	DAR0	RW	0x--------	32-bit	Direct Access Register 0
0x4	DAR1	RW	0x--------	32-bit	Direct Access Register 1
0x8	DAR2	RW	0x--------	32-bit	Direct Access Register 2
...	...				
0x3fc	DAR255	RW	0x--------	32-bit	Direct Access Register 255
0xd00	CSW	RW	0x30-060--	32-bit	Control Status Word register
0xd04	TAR	RW	0x00000000	32-bit	Transfer Address Register
0xd08	TARH	RW	0x00000000	32-bit	Transfer Address Register
0xd0c	DRW	RW	0x--------	32-bit	Data Read/Write register
0xd10	BD0	RW	0x--------	32-bit	Banked Data register 0
0xd14	BD1	RW	0x--------	32-bit	Banked Data register 1

Offset	Name	Type	Reset	Width	Description
0xd18	BD2	RW	0x--------	32-bit	Banked Data register 2
0xd1c	BD3	RW	0x--------	32-bit	Banked Data register 3
0xd24	TRR	RW	0x00000000	32-bit	Transfer Response Register
0xddc	MECID	RW	0x00000000	32-bit	Memory Encryption Contexts Register
0xde0	CFG1	RO	0x00000000	32-bit	Configuration Register 1
0xdf0	BASEH	RW	0x--------	32-bit	Debug Base Address register upper 32 bits
0xdf4	CFG	RO	0x010101A-	32-bit	Configuration register
0xdf8	BASE	RO	0x-----00-	32-bit	Debug Base Address register
0xdfc	IDR	RO	0x64770017	32-bit	Identification Register
0xefc	ITSTATUS	RO	0x00000000	32-bit	Integration Test Status register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x00000003	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x0-00-0--	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x47700A17	32-bit	Device Architecture Register
0xfcc	DEVTYPE	RO	0x00000000	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000E4	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000006B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.12.1 css600_axiap Direct Access Register 0, DAR0

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x0

Type

RW

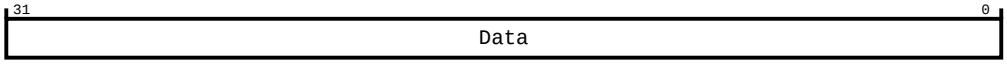
Reset value

0x--------

Bit descriptions

The following figure shows the DAR0 register bit assignments.

Figure 9-269: Bit assignment diagram for the DAR0 register



The following table shows the DAR0 register bit descriptions.

Table 9-280: DAR0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFC00) + 0x0). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.12.2 css600_axiap Direct Access Register 1, DAR1

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x4

Type

RW

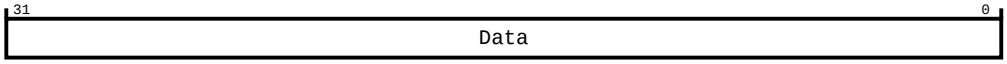
Reset value

0x--------

Bit descriptions

The following figure shows the DAR1 register bit assignments.

Figure 9-270: Bit assignment diagram for the DAR1 register



The following table shows the DAR1 register bit descriptions.

Table 9-281: DAR1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFC00`) + `0x4`). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.12.3 css600_axiap Direct Access Register 2, DAR2

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x8`

Type

RW

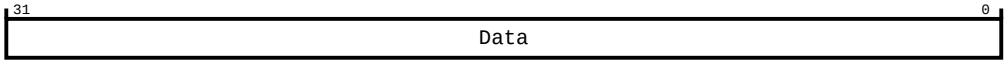
Reset value

`0x--------`

Bit descriptions

The following figure shows the DAR2 register bit assignments.

Figure 9-271: Bit assignment diagram for the DAR2 register



The following table shows the DAR2 register bit descriptions.

Table 9-282: DAR2 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFC00) + 0x8). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.12.4 css600_axiap Direct Access Register 255, DAR255

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x3FC

Type

RW

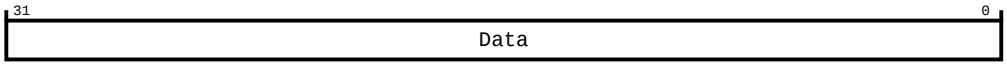
Reset value

0x--------

Bit descriptions

The following figure shows the DAR255 register bit assignments.

Figure 9-272: Bit assignment diagram for the DAR255 register



The following table shows the DAR255 register bit descriptions.

Table 9-283: DAR255 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFC00) + 0x3FC). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.12.5 css600_axiap Control Status Word register, CSW

The CSW register configures and controls accesses through the Mem-AP to the connected memory system.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD00`

Type

RW

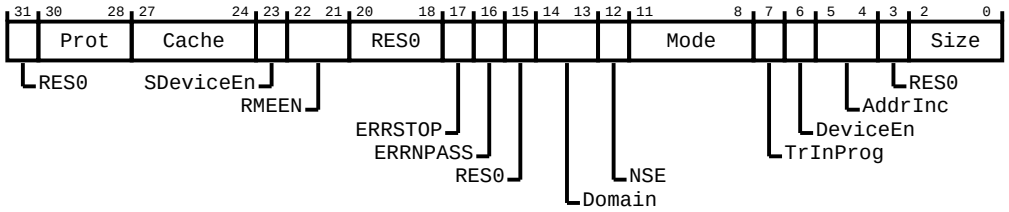
Reset value

`0x30-060--`

Bit descriptions

The following figure shows the CSW register bit assignments.

Figure 9-273: Bit assignment diagram for the CSW register



The following table shows the CSW register bit descriptions.

Table 9-284: CSW bit descriptions

Bits	Name	Reset	Type	Description
31	RES0	`0b0`	RO	Reserved bit or field with SBZP behavior.
30:28	Prot	`0b011`	RW	Drives AXI manager interface ports arprot_m[2:0] and awprot_m[2:0] which specifies the protection encoding. The reset value is `0x3` (Data, Non-secure, Privileged). Together with CSW.NSE, CSW.Prot[1] determines the PAS of the access to be initiated by the Access Port: Non-Secure, Secure, Realm or Root. An access will only be initiated if permitted by CSW.DeviceEn, CSW.SDeviceEn and CSW.RMEEN.

Bits	Name	Reset	Type	Description
27:24	Cache	0b0000	RW	Specifies the AXI cache encoding. Software must never program an invalid combination of values in CSW.Cache and CSW.Domain fields. The software must use different cache encoding values for reads and writes. If an illegal set of cache and domain values is programmed, the AXI-AP does not issue the transaction on its manager interface and generates a memory access error.
23	SDeviceEn	**UNKNOWN**	RO	Secure Debug Enabled. This field has one of the following values: **0b0** Secure access is disabled. **0b1** Secure access is enabled.
22:21	RMEEN	**UNKNOWN**	RO	Realm and Root access status. When legacy_tz_en==1 this field reads 0b00. **0b00** Realm and Root access is disabled. **0b01** Realm access is enabled, Root access is disabled. **0b10** Reserved. **0b11** Realm and Root access is enabled.
20:18	**RES0**	0b000	RO	Reserved bit or field with SBZP behavior.
17	ERRSTOP	0b0	RW	Stop on error. **0b0** Memory access errors do not prevent future memory accesses. **0b1** Memory access errors prevent future memory accesses.
16	ERRNPASS	0b0	RW	Errors are not passed upstream. **0b0** Memory access errors are passed upstream. **0b1** Memory access errors are not passed upstream.
15	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
14:13	Domain	0b11	RW	Shareable transaction encoding defined in the AMBA AXI Protocol Specification. **0b00** Non-shareable. **0b01** Shareable, inner domain, includes additional managers. **0b10** Shareable, outer domain, also includes inner or additional managers. **0b11** Shareable, system domain, all managers included.
12	NSE	0b0	RW	Select Root and Realm. When legacy_tz_en==1 this field reads 0b0. **0b0** CSW.Prot[1] selects Secure or Non-Secure PAS. **0b1** CSW.Prot[1] selects Root or Realm PAS.
11:8	Mode	0b0000	RO	Mode of operation of the Mem-AP. **0b0000** Basic (normal download or upload) mode.
7	TrInProg	0b0	RO	Transfer in progress. After an ABORT operation, debug software can read this bit to check whether the aborted transaction completed. **0b0** The connection to the memory system is idle. **0b1** A transfer is in progress on the connection to the memory system.
6	DeviceEn	**UNKNOWN**	RO	Device enabled. **0b0** The Mem-AP is not enabled. **0b1** Transactions can be issued through the Mem-AP.
5:4	AddrInc	0b00	RW	Auto address increment mode on RW data access. Only increments if the current transaction completes without an error response and the transaction is not aborted. **0b00** Address auto-increment disabled. **0b01** Address increment-single enabled. **0b10** Reserved. **0b11** Reserved.

Bits	Name	Reset	Type	Description
3	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
2:0	Size	**IMPLEMENTATION DEFINED**	RW	Size of the data access to perform. **0b000** 8 bits. **0b001** 16 bits. **0b010** 32 bits. **0b011** 64 bits, if LDE is supported, Reserved, if LDE is not supported. **0b100** Reserved. **0b101** Reserved. **0b110** Reserved. **0b111** Reserved.

9.12.6 css600_axiap Transfer Address Register, TAR

TAR holds the transfer address of the current transfer. TAR must be programmed before initiating any memory transfer through DRW, or Banked Data Registers, or Direct Access Registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD04

Type

RW

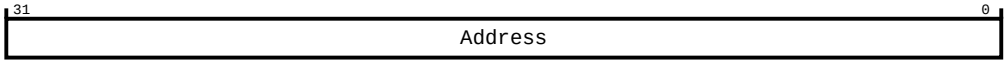
Reset value

0x00000000

Bit descriptions

The following figure shows the TAR register bit assignments.

Figure 9-274: Bit assignment diagram for the TAR register



The following table shows the TAR register bit descriptions.

Table 9-285: TAR bit descriptions

Bits	Name	Reset	Type	Description
31:0	Address	0x0	RW	Address of the current transfer. When a memory access is initiated by accessing the DRW register, the TAR value directly gives the 32-bit transfer address. When a memory access is initiated by accessing Banked Data registers, the TAR only provides the upper bits [31:4] and the remaining address bits [3:0] come from the offset of Banked Data register being accessed. When a memory access is initiated by accessing Direct Access Registers, the TAR provides the upper bits [31:10] and the remaining address bits [9:0] come from the offset of the DAR being accessed.

9.12.7 css600_axiap Transfer Address Register, TARH

The TARH register holds the upper 32 bits of the Transfer Address Register if the Mem-AP supports LAE.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD08

Type

RW

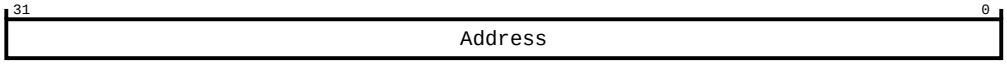
Reset value

0x00000000

Bit descriptions

The following figure shows the TARH register bit assignments.

Figure 9-275: Bit assignment diagram for the TARH register



The following table shows the TARH register bit descriptions.

Table 9-286: TARH bit descriptions

Bits	Name	Reset	Type	Description
31:0	Address	0x0	RW	Bits [63:32] of the transfer address. This register is present only when LAE is supported, otherwise it is reserved and **RAZ/WI**.

9.12.8 css600_axiap Data Read/Write register, DRW

A write to the DRW register initiates a memory write transaction. AP drives DRW write data on the data bus during the data phase of the current transfer. Reading the DRW register initiates a memory read transaction.

The resulting read data that is received from the memory system is returned on the completer interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD0C

Type

RW

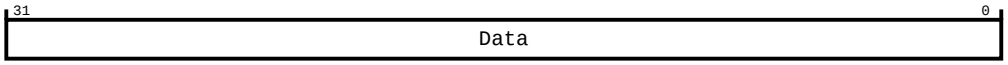
Reset value

0x--------

Bit descriptions

The following figure shows the DRW register bit assignments.

Figure 9-276: Bit assignment diagram for the DRW register



The following table shows the DRW register bit descriptions.

Table 9-287: DRW bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Current transfer data value. Writing to this register initiates a write to the address specified by the TAR. Reading this register initiates a read from the address specified by the TAR. When the read completes, the data value is returned in this register.

9.12.9 css600_axiap Banked Data register 0, BD0

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD10`

Type

RW

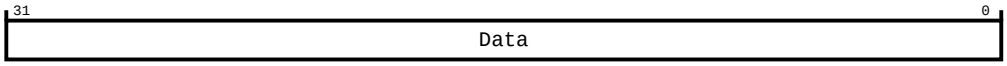
Reset value

`0x--------`

Bit descriptions

The following figure shows the BD0 register bit assignments.

Figure 9-277: Bit assignment diagram for the BD0 register



The following table shows the BD0 register bit descriptions.

Table 9-288: BD0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFFF0`) + `0x0`). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.12.10 css600_axiap Banked Data register 1, BD1

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD14`

Type

RW

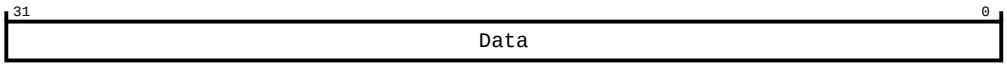
Reset value

`0x--------`

Bit descriptions

The following figure shows the BD1 register bit assignments.

Figure 9-278: Bit assignment diagram for the BD1 register



The following table shows the BD1 register bit descriptions.

Table 9-289: BD1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFFF0`) + `0x4`). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.12.11 css600_axiap Banked Data register 2, BD2

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD18

Type

RW

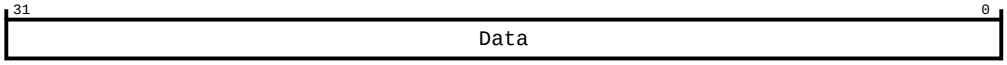
Reset value

0x--------

Bit descriptions

The following figure shows the BD2 register bit assignments.

Figure 9-279: Bit assignment diagram for the BD2 register



The following table shows the BD2 register bit descriptions.

Table 9-290: BD2 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFFF0) + 0x8). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.12.12 css600_axiap Banked Data register 3, BD3

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD1C

Type

RW

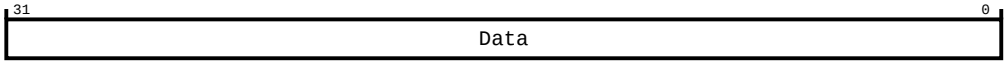
Reset value

0x--------

Bit descriptions

The following figure shows the BD3 register bit assignments.

Figure 9-280: Bit assignment diagram for the BD3 register



The following table shows the BD3 register bit descriptions.

Table 9-291: BD3 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFFF0) + 0xC). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.12.13 css600_axiap Transfer Response Register, TRR

The Transfer Response Register is used to capture an error response received during a transaction. It is also used to clear any logged responses.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD24

Type

RW

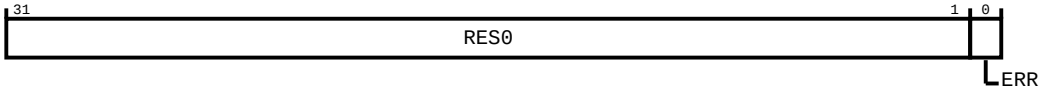
Reset value

0x00000000

Bit descriptions

The following figure shows the TRR register bit assignments.

Figure 9-281: Bit assignment diagram for the TRR register



The following table shows the TRR register bit descriptions.

Table 9-292: TRR bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	ERR	0b0	RW	Logged error. **0b0** On reads - no error response logged. Writing to this bit has no effect. **0b1** On reads - error response logged. Writing to this bit clears this bit to 0.

9.12.14 css600_axiap Memory Encryption Contexts Register, MECID

The MECID register holds the memory encryption context ID to be used for the next memory transaction.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDDC

Type

RW

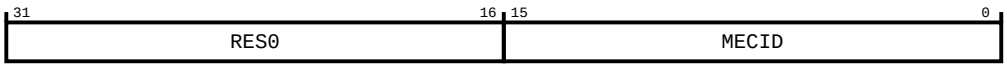
Reset value

0x00000000

Bit descriptions

The following figure shows the MECID register bit assignments.

Figure 9-282: Bit assignment diagram for the MECID register



The following table shows the MECID register bit descriptions.

Table 9-293: MECID bit descriptions

Bits	Name	Reset	Type	Description
31:16	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
15:0	MECID	0x0	RW	MECID value to output on the AXI manager interface.

9.12.15 css600_axiap Configuration Register 1, CFG1

The CFG1 register returns details of the MTE support implemented in the Mem-AP.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDE0

Type

RO

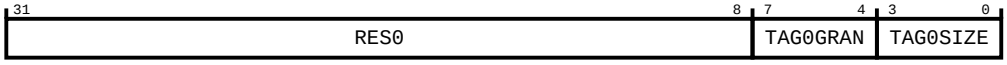
Reset value

0x00000000

Bit descriptions

The following figure shows the CFG1 register bit assignments.

Figure 9-283: Bit assignment diagram for the CFG1 register



The following table shows the CFG1 register bit descriptions.

Table 9-294: CFG1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	TAG0GRAN	0b0000	RO	Memory tagging granule.
3:0	TAG0SIZE	0b0000	RO	Memory tagging support. **0b0000** Memory tagging is not implemented.

9.12.16 css600_axiap Debug Base Address register upper 32 bits, BASEH

The BASEH register holds the upper 32 bits of the base address of a ROM table when LAE is supported.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xDF0`

Type

RW

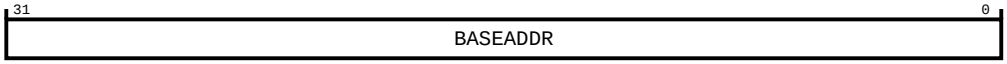
Reset value

`0x--------`

Bit descriptions

The following figure shows the BASEH register bit assignments.

Figure 9-284: Bit assignment diagram for the BASEH register



The following table shows the BASEH register bit descriptions.

Table 9-295: BASEH bit descriptions

Bits	Name	Reset	Type	Description
31:0	BASEADDR	**IMPLEMENTATION DEFINED**	RO	Bits [63:32] of the ROM table base address. This register is present only when LAE is supported, otherwise it is reserved and **RAZ/WI**. Even with LAE supported, this field is valid only if BASE.EntryPresent bit is set to 1, in which case it returns the tie-off value of the input signal baseaddr[63:32]. Otherwise, it reads as `0x0`.

9.12.17 css600_axiap Configuration register, CFG

This is the AXI-AP configuration register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDF4

Type

RO

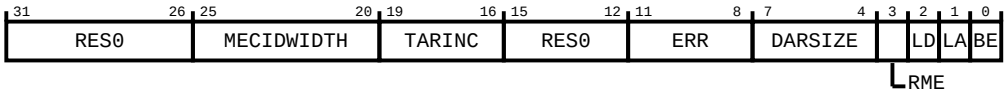
Reset value

0x010101A-

Bit descriptions

The following figure shows the CFG register bit assignments.

Figure 9-285: Bit assignment diagram for the CFG register



The following table shows the CFG register bit descriptions.

Table 9-296: CFG bit descriptions

Bits	Name	Reset	Type	Description
31:26	**RES0**	0b000000	RO	Reserved bit or field with SBZP behavior.
25:20	MECIDWIDTH	0b010000	RO	Memory Encryption Context (MEC) ID width. **0b010000** 16-bit MECID width.
19:16	TARINC	0b0001	RO	TAR incrementer size. **0b0001** TAR incrementer is 10-bits.
15:12	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
11:8	ERR	0b0001	RO	Identifies the type of error handling that is implemented. **0b0001** TRR, CSW.ERRNPASS and CSW.ERRSTOP are implemented.
7:4	DARSIZE	0b1010	RO	Size of DAR register space. **0b1010** DAR0-DAR255 are implemented.
3	RME	**IMPLEMENTATION DEFINED**	RO	Realm Management Extension **0b0** Realm Management Extension not implemented. **0b1** Realm Management Extension implemented.

Bits	Name	Reset	Type	Description
2	LD	**IMPLEMENTATION DEFINED**	RO	Large Data. Indicates support for LDE (data items greater than 32 bits). **0b0** Mem-AP does not support data items that are larger than 32 bits. **0b1** Mem-AP implements the Large Data Extension, and supports data items larger than 32-bits.
1	LA	**IMPLEMENTATION DEFINED**	RO	Long Address. Indicates support for LAE (greater than 32-bit of addressing). **0b0** Mem-AP only supports physical addresses of 32 bits or smaller. Registers `0xD08` and `0xDF0` are reserved. **0b1** Mem-AP supports physical addresses with more than 32 bits. TAR and BASE registers occupy two locations, at `0xD04` and `0xD08`, and at `0xDF8` and `0xDF0` respectively.
0	BE	`0b0`	RO	Big-endian. Always read as 0, BE support is obsolete in Mem-AP from ADIv5.2 **0b0** Not supported.

9.12.18 css600_axiap Debug Base Address register, BASE

Provides an initial system address for the first component in the system. Typically, the system address is the address of a top-level ROM Table that indicates where APv2 APs are located.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xDF8`

Type

RO

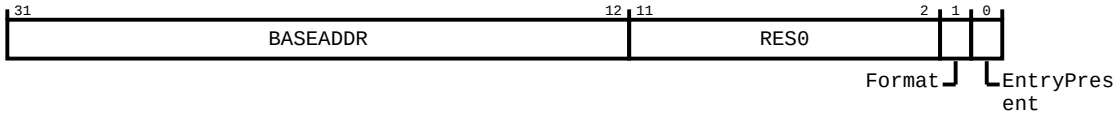
Reset value

`0x-----00-`

Bit descriptions

The following figure shows the BASE register bit assignments.

Figure 9-286: Bit assignment diagram for the BASE register



The following table shows the BASE register bit descriptions.

Table 9-297: BASE bit descriptions

Bits	Name	Reset	Type	Description
31:12	BASEADDR	**IMPLEMENTATION DEFINED**	RO	Base address of a ROM table. It points to the start of the debug register space or a ROM table address. Bits[11:0] of the address are `0x000` because the address is aligned to 4KB boundary. This field is valid only if BASE.EntryPresent bit is set to 1, in which case it returns the tie-off value of the input signal baseaddr[31:12], otherwise, it reads as `0x0`.
11:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1	Format	`0b1`	RO	Base address register format. Returns the value `0b1` indicating the ADIv5 format, which is unchanged in ADIv6.
0	EntryPresent	**IMPLEMENTATION DEFINED**	RO	This field indicates whether a debug component is present for this AP. It returns the tie-off value of the input signal baseaddr_valid. **`0b0`** No debug entry present. **`0b1`** Debug entry present and BASE.BASEADDR indicate the start address of the debug register space or ROM table.

9.12.19 css600_axiap Identification Register, IDR

This register provides a mechanism for the debugger to know various identity attributes of the AP.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xDFC`

Type

RO

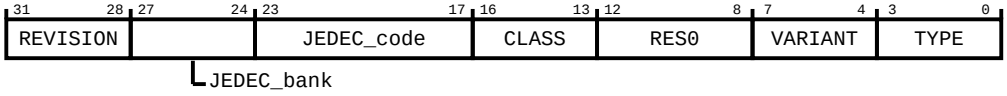
Reset value

`0x64770017`

Bit descriptions

The following figure shows the IDR register bit assignments.

Figure 9-287: Bit assignment diagram for the IDR register



The following table shows the IDR register bit descriptions.

Table 9-298: IDR bit descriptions

Bits	Name	Reset	Type	Description
31:28	REVISION	0b0110	RO	Revision. An incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
27:24	JEDEC_bank	0b0100	RO	The JEP106 continuation code. **0b0100** Arm
23:17	JEDEC_code	0x3B	RO	The JEP106 identification code. **0x3B** Arm
16:13	CLASS	0b1000	RO	Defines the class of the AP. **0b1000** MEM-AP
12:8	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
7:4	VARIANT	0b0001	RO	Together with the TYPE field, this field identifies the AP implementation. VARIANT differentiates AP implementations that have the same value of TYPE.
3:0	TYPE	0b0111	RO	Indicates the type of bus, or other connection, that connects to the AP. **0b0111** AMBA AXI5 bus.

9.12.20 css600_axiap Integration Test Status register, ITSTATUS

This register indicates the Integration Test DP Abort status.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEFC

Type

RO

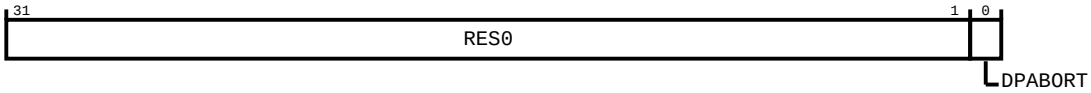
Reset value

0x00000000

Bit descriptions

The following figure shows the ITSTATUS register bit assignments.

Figure 9-288: Bit assignment diagram for the ITSTATUS register



The following table shows the ITSTATUS register bit descriptions.

Table 9-299: ITSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	DPABORT	0b0	RO	When in Integration testing mode (ITCTRL.IME=0b1): Behaves as a sticky bit and latches to 1 on a rising edge of dp_abort. Cleared on a read from this register. If dp_abort rises in the same cycle as a read of the ITSTATUS register is received, the read takes priority and the register is cleared. When in normal functional operation mode (ITCTRL.IME=0b0): Read as 0, writes ignored.

9.12.21 css600_axiap Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

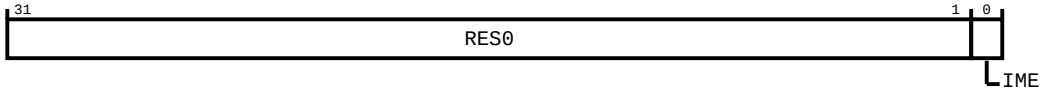
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-289: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-300: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.12.22 css600_axiap Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

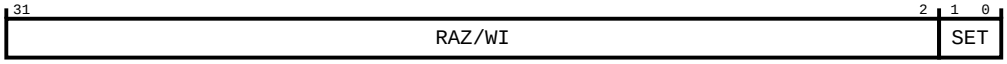
Reset value

0x00000003

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-290: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-301: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RAZ/WI**	0x0	RO	**RAZ/WI**.
1:0	SET	0b11	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.12.23 css600_axiap Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA4

Type

RW

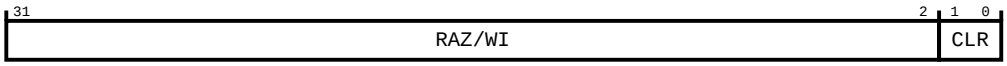
Reset value

0x00000000

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-291: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-302: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RAZ/WI**	0x0	RO	**RAZ/WI.**
1:0	CLR	0b00	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.12.24 css600_axiap Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

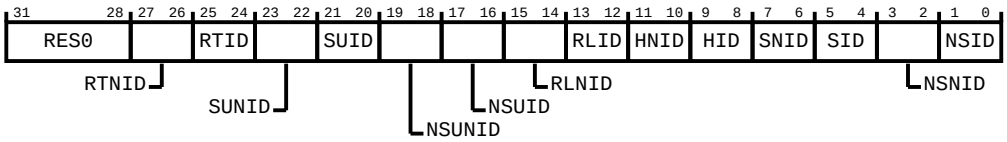
Reset value

0x0-00-0--

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-292: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-303: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
27:26	RTNID	**UNKNOWN**	RO	Root non-invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
25:24	RTID	**UNKNOWN**	RO	Root invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	**UNKNOWN**	RO	Realm non-invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
13:12	RLID	**UNKNOWN**	RO	Realm invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
7:6	SNID	**UNKNOWN**	RO	Secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
5:4	SID	**UNKNOWN**	RO	Secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
3:2	NSNID	**UNKNOWN**	RO	Non-secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
1:0	NSID	**UNKNOWN**	RO	Non-secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.

9.12.25 css600_axiap Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFBC`

Type

RO

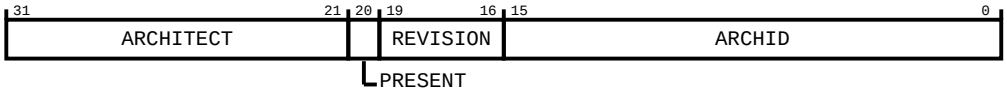
Reset value

`0x47700A17`

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-293: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-304: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x23B	RO	Defines the architect of the component **0x23B** Arm
20	PRESENT	0b1	RO	Indicates the presence of this register **0b1** DEVARCH is present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0A17	RO	Architecture ID. Returns a value that identifies the architecture of the component. **0x0A17** Memory Access Port v2 architecture

9.12.26 css600_axiap Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

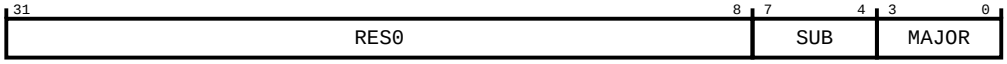
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-294: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-305: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	0b0000	RO	Minor classification. Returns 0x0, Other/undefined.
3:0	MAJOR	0b0000	RO	Major classification. Returns 0x0, Miscellaneous.

9.12.27 css600_axiap Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

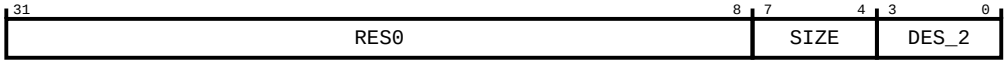
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-295: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-306: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.12.28 css600_axiap Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

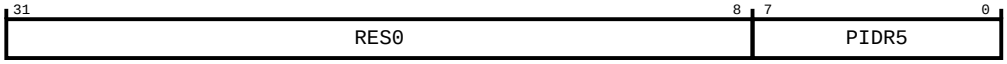
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-296: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-307: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.12.29 css600_axiap Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

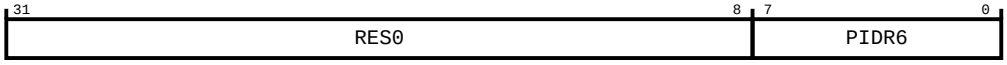
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-297: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-308: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.12.30 css600_axiap Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFDC`

Type

RO

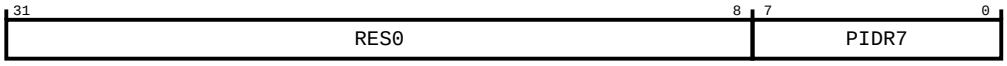
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-298: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-309: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.12.31 css600_axiap Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

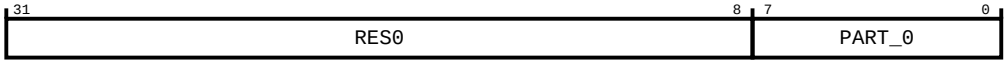
Reset value

0x000000E4

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-299: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-310: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xE4	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.12.32 css600_axiap Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

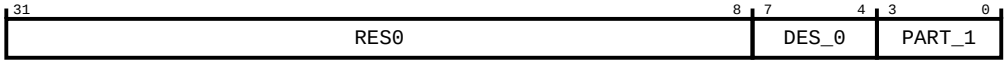
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-300: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-311: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.12.33 css600_axiap Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

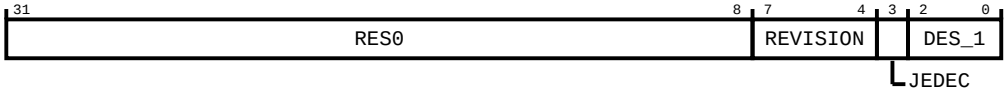
Reset value

0x0000006B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-301: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-312: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b0110	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.12.34 css600_axiap Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

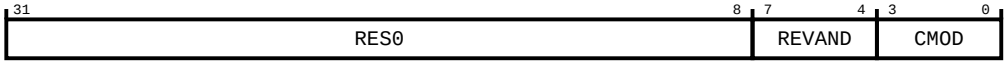
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-302: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-313: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	`0b0000`	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is `0x0`.
3:0	CMOD	`0b0000`	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is `0x0`.

9.12.35 css600_axiap Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

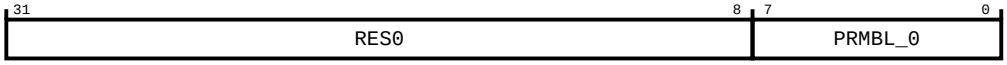
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-303: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-314: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.12.36 css600_axiap Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

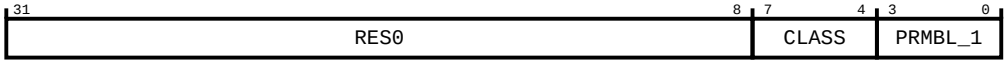
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-304: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-315: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	`0b1001`	RO	Component class **`0b1001`** CoreSight component
3:0	PRMBL_1	`0b0000`	RO	Preamble. Returns `0x0`.

9.12.37 css600_axiap Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF8`

Type

RO

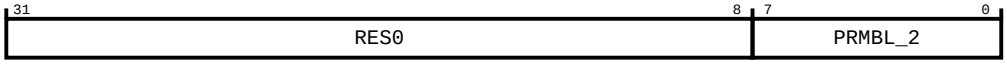
Reset value

`0x00000005`

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-305: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-316: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.12.38 css600_axiap Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

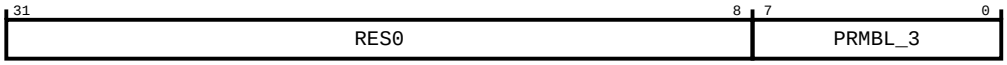
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-306: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-317: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.13 css600_axiap_mte register summary

This section describes the css600_axiap_mte_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-318: css600_axiap_mte_registers register summary

Offset	Name	Type	Reset	Width	Description
0x0	DAR0	RW	0x--------	32-bit	Direct Access Register 0
0x4	DAR1	RW	0x--------	32-bit	Direct Access Register 1
0x8	DAR2	RW	0x--------	32-bit	Direct Access Register 2
...	...				
0x3fc	DAR255	RW	0x--------	32-bit	Direct Access Register 255
0xd00	CSW	RW	0x30-060-3	32-bit	Control Status Word register
0xd04	TAR	RW	0x00000000	32-bit	Transfer Address Register
0xd08	TARH	RW	0x00000000	32-bit	Transfer Address Register
0xd0c	DRW	RW	0x--------	32-bit	Data Read/Write register
0xd10	BD0	RW	0x--------	32-bit	Banked Data register 0
0xd14	BD1	RW	0x--------	32-bit	Banked Data register 1
0xd18	BD2	RW	0x--------	32-bit	Banked Data register 2
0xd1c	BD3	RW	0x--------	32-bit	Banked Data register 3
0xd24	TRR	RW	0x00000000	32-bit	Transfer Response Register
0xd30	T0TR	RW	0x00000000	32-bit	Tag 0 Transfer Register
0xddc	MECID	RW	0x00000000	32-bit	Memory Encryption Contexts Register
0xde0	CFG1	RO	0x00000044	32-bit	Configuration Register 1
0xdf0	BASEH	RW	0x--------	32-bit	Debug Base Address register upper 32 bits
0xdf4	CFG	RO	0x010101A-	32-bit	Configuration register
0xdf8	BASE	RO	0x-----00-	32-bit	Debug Base Address register
0xdfc	IDR	RO	0x64770017	32-bit	Identification Register
0xefc	ITSTATUS	RO	0x00000000	32-bit	Integration Test Status register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x00000003	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x0-00-0--	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x47700A17	32-bit	Device Architecture Register
0xfcc	DEVTYPE	RO	0x00000000	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000E4	32-bit	Peripheral Identification Register 0

Offset	Name	Type	Reset	Width	Description
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000006B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.13.1 css600_axiap_mte Direct Access Register 0, DAR0

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x0

Type

RW

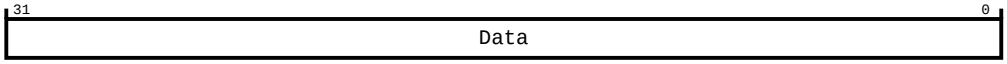
Reset value

0x--------

Bit descriptions

The following figure shows the DAR0 register bit assignments.

Figure 9-307: Bit assignment diagram for the DAR0 register



The following table shows the DAR0 register bit descriptions.

Table 9-319: DAR0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFC00) + 0x0). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.13.2 css600_axiap_mte Direct Access Register 1, DAR1

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x4

Type

RW

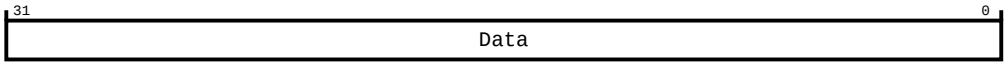
Reset value

0x--------

Bit descriptions

The following figure shows the DAR1 register bit assignments.

Figure 9-308: Bit assignment diagram for the DAR1 register



The following table shows the DAR1 register bit descriptions.

Table 9-320: DAR1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFC00) + 0x4). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.13.3 css600_axiap_mte Direct Access Register 2, DAR2

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x8`

Type

RW

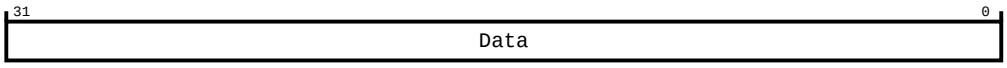
Reset value

`0x--------`

Bit descriptions

The following figure shows the DAR2 register bit assignments.

Figure 9-309: Bit assignment diagram for the DAR2 register



The following table shows the DAR2 register bit descriptions.

Table 9-321: DAR2 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFC00`) + `0x8`). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.13.4 css600_axiap_mte Direct Access Register 255, DAR255

The Direct Access Registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 1kB boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x3FC`

Type

RW

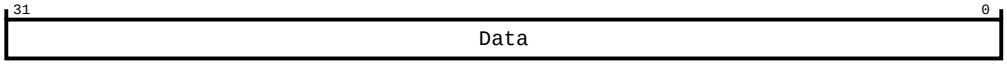
Reset value

`0x--------`

Bit descriptions

The following figure shows the DAR255 register bit assignments.

Figure 9-310: Bit assignment diagram for the DAR255 register



The following table shows the DAR255 register bit descriptions.

Table 9-322: DAR255 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFC00`) + `0x3FC`). Writing to this register initiates a write to the address specified by the TAR and the DAR register. Reading this register initiates a read from the address specified by the TAR and the DAR register. When the read completes, the data value is returned in this register.

9.13.5 css600_axiap_mte Control Status Word register, CSW

The CSW register configures and controls accesses through the Mem-AP to the connected memory system.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD00`

Type

RW

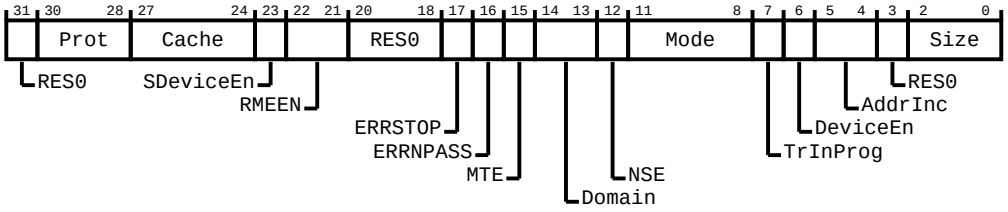
Reset value

`0x30-060-3`

Bit descriptions

The following figure shows the CSW register bit assignments.

Figure 9-311: Bit assignment diagram for the CSW register



The following table shows the CSW register bit descriptions.

Table 9-323: CSW bit descriptions

Bits	Name	Reset	Type	Description
31	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
30:28	Prot	`0b011`	RW	Drives AXI manager interface ports arprot_m[2:0] and awprot_m[2:0] which specifies the protection encoding. The reset value is `0x3` (Data, Non-secure, Privileged). Together with CSW.NSE, CSW.Prot[1] determines the PAS of the access to be initiated by the Access Port: Non-Secure, Secure, Realm or Root. An access will only be initiated if permitted by CSW.DeviceEn, CSW.SDeviceEn and CSW.RMEEN.
27:24	Cache	`0b0000`	RW	Specifies the AXI cache encoding. Software must never program an invalid combination of values in CSW.Cache and CSW.Domain fields. The software must use different cache encoding values for reads and writes. If an illegal set of cache and domain values is programmed, the AXI-AP does not issue the transaction on its manager interface and generates a memory access error.
23	SDeviceEn	**UNKNOWN**	RO	Secure Debug Enabled. This field has one of the following values: **`0b0`** Secure access is disabled. **`0b1`** Secure access is enabled.

Bits	Name	Reset	Type	Description
22:21	RMEEN	**UNKNOWN**	RO	Realm and Root access status. When legacy_tz_en==1 this field reads `0b00`. **0b00** Realm and Root access is disabled. **0b01** Realm access is enabled, Root access is disabled. **0b10** Reserved. **0b11** Realm and Root access is enabled.
20:18	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
17	ERRSTOP	`0b0`	RW	Stop on error. **0b0** Memory access errors do not prevent future memory accesses. **0b1** Memory access errors prevent future memory accesses.
16	ERRNPASS	`0b0`	RW	Errors are not passed upstream. **0b0** Memory access errors are passed upstream. **0b1** Memory access errors are not passed upstream.
15	MTE	`0b0`	RW	Memory Tagging control. When memory tagging accesses are enabled, system read and write accesses via DRW, BD0-BD3, and DAR0-DAR255, use T0TR for transferring tag information. Software must set CSW.Size=64bit when the MTE bit is set, otherwise the Mem-AP does not issue a transaction on its manager interface and generates a memory access error. When Memory Tagging support is not implemented this register field is reserved and access type is **RAZ/WI**. **0b0** Memory Tagging accesses disabled. **0b1** Memory Tagging accesses enabled.
14:13	Domain	`0b11`	RW	Shareable transaction encoding defined in the AMBA AXI Protocol Specification. **0b00** Non-shareable. **0b01** Shareable, inner domain, includes additional managers. **0b10** Shareable, outer domain, also includes inner or additional managers. **0b11** Shareable, system domain, all managers included.

Bits	Name	Reset	Type	Description
12	NSE	0b0	RW	Select Root and Realm. When legacy_tz_en==1 this field reads 0b0. **0b0** CSW.Prot[1] selects Secure or Non-Secure PAS. **0b1** CSW.Prot[1] selects Root or Realm PAS.
11:8	Mode	0b0000	RO	Mode of operation of the Mem-AP. **0b0000** Basic (normal download or upload) mode.
7	TrInProg	0b0	RO	Transfer in progress. After an ABORT operation, debug software can read this bit to check whether the aborted transaction completed. **0b0** The connection to the memory system is idle. **0b1** A transfer is in progress on the connection to the memory system.
6	DeviceEn	UNKNOWN	RO	Device enabled. **0b0** The Mem-AP is not enabled. **0b1** Transactions can be issued through the Mem-AP.
5:4	AddrInc	0b00	RW	Auto address increment mode on RW data access. Only increments if the current transaction completes without an error response and the transaction is not aborted. **0b00** Address auto-increment disabled. **0b01** Address increment-single enabled. **0b10** Reserved. **0b11** Reserved.
3	RES0	0b0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
2:0	Size	0b011	RW	Size of the data access to perform. **0b000** 8 bits. **0b001** 16 bits. **0b010** 32 bits. **0b011** 64 bits, if LDE is supported, Reserved, if LDE is not supported. **0b100** Reserved. **0b101** Reserved. **0b110** Reserved. **0b111** Reserved.

9.13.6 css600_axiap_mte Transfer Address Register, TAR

TAR holds the transfer address of the current transfer. TAR must be programmed before initiating any memory transfer through DRW, or Banked Data Registers, or Direct Access Registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD04

Type

RW

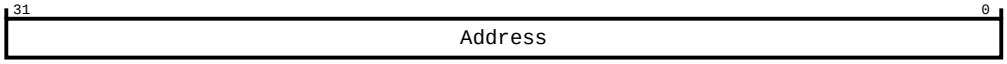
Reset value

0x00000000

Bit descriptions

The following figure shows the TAR register bit assignments.

Figure 9-312: Bit assignment diagram for the TAR register



The following table shows the TAR register bit descriptions.

Table 9-324: TAR bit descriptions

Bits	Name	Reset	Type	Description
31:0	Address	0x0	RW	Address of the current transfer. When a memory access is initiated by accessing the DRW register, the TAR value directly gives the 32-bit transfer address. When a memory access is initiated by accessing Banked Data registers, the TAR only provides the upper bits [31:4] and the remaining address bits [3:0] come from the offset of Banked Data register being accessed. When a memory access is initiated by accessing Direct Access Registers, the TAR provides the upper bits [31:10] and the remaining address bits [9:0] come from the offset of the DAR being accessed.

9.13.7 css600_axiap_mte Transfer Address Register, TARH

The TARH register holds the upper 32 bits of the Transfer Address Register if the Mem-AP supports LAE.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD08

Type

RW

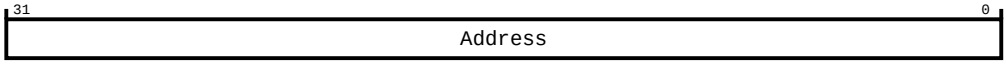
Reset value

0x00000000

Bit descriptions

The following figure shows the TARH register bit assignments.

Figure 9-313: Bit assignment diagram for the TARH register



The following table shows the TARH register bit descriptions.

Table 9-325: TARH bit descriptions

Bits	Name	Reset	Type	Description
31:0	Address	0x0	RW	Bits [63:32] of the transfer address. This register is present only when LAE is supported, otherwise it is reserved and **RAZ/WI**.

9.13.8 css600_axiap_mte Data Read/Write register, DRW

A write to the DRW register initiates a memory write transaction. AP drives DRW write data on the data bus during the data phase of the current transfer. Reading the DRW register initiates a memory read transaction.

The resulting read data that is received from the memory system is returned on the completer interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD0C

Type

RW

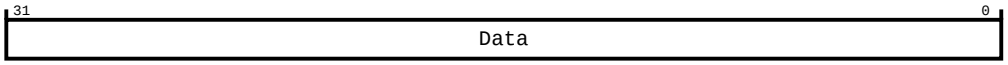
Reset value

0x--------

Bit descriptions

The following figure shows the DRW register bit assignments.

Figure 9-314: Bit assignment diagram for the DRW register



The following table shows the DRW register bit descriptions.

Table 9-326: DRW bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Current transfer data value. Writing to this register initiates a write to the address specified by the TAR. Reading this register initiates a read from the address specified by the TAR. When the read completes, the data value is returned in this register.

9.13.9 css600_axiap_mte Banked Data register 0, BD0

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD10`

Type

RW

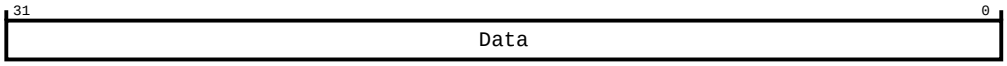
Reset value

`0x--------`

Bit descriptions

The following figure shows the BD0 register bit assignments.

Figure 9-315: Bit assignment diagram for the BD0 register



The following table shows the BD0 register bit descriptions.

Table 9-327: BD0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFFF0`) + `0x0`). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.13.10 css600_axiap_mte Banked Data register 1, BD1

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD14`

Type

RW

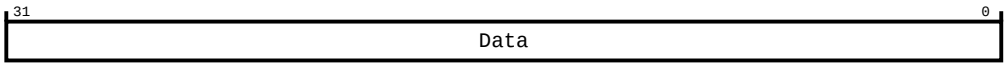
Reset value

`0x--------`

Bit descriptions

The following figure shows the BD1 register bit assignments.

Figure 9-316: Bit assignment diagram for the BD1 register



The following table shows the BD1 register bit descriptions.

Table 9-328: BD1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & `0xFFFFFFF0`) + `0x4`). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.13.11 css600_axiap_mte Banked Data register 2, BD2

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD18

Type

RW

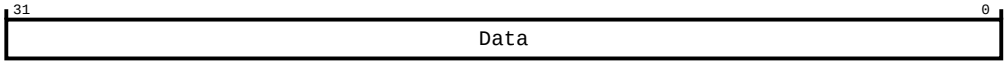
Reset value

0x--------

Bit descriptions

The following figure shows the BD2 register bit assignments.

Figure 9-317: Bit assignment diagram for the BD2 register



The following table shows the BD2 register bit descriptions.

Table 9-329: BD2 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFFF0) + 0x8). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.13.12 css600_axiap_mte Banked Data register 3, BD3

The Banked Data registers provide a mechanism for directly mapping APB completer accesses to memory transfers without having to rewrite the TAR within a 16-byte boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD1C

Type

RW

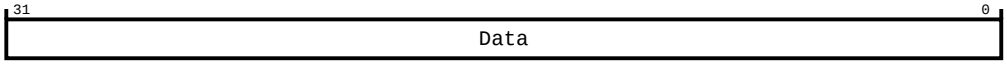
Reset value

0x--------

Bit descriptions

The following figure shows the BD3 register bit assignments.

Figure 9-318: Bit assignment diagram for the BD3 register



The following table shows the BD3 register bit descriptions.

Table 9-330: BD3 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Data	UNKNOWN	RW	Maps to memory address ((TAR & 0xFFFFFFF0) + 0xC). Writing to this register initiates a write to the address specified by the TAR and the BD register. Reading this register initiates a read from the address specified by the TAR and the BD register. When the read completes, the data value is returned in this register.

9.13.13 css600_axiap_mte Transfer Response Register, TRR

The Transfer Response Register is used to capture an error response received during a transaction. It is also used to clear any logged responses.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD24

Type

RW

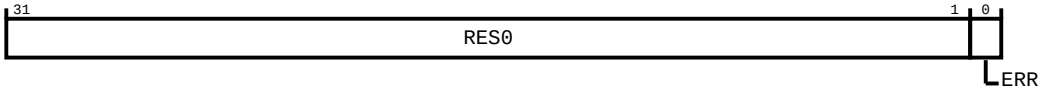
Reset value

0x00000000

Bit descriptions

The following figure shows the TRR register bit assignments.

Figure 9-319: Bit assignment diagram for the TRR register



The following table shows the TRR register bit descriptions.

Table 9-331: TRR bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	ERR	0b0	RW	Logged error. **0b0** On reads - no error response logged. Writing to this bit has no effect. **0b1** On reads - error response logged. Writing to this bit clears this bit to 0.

9.13.14 css600_axiap_mte Tag 0 Transfer Register, T0TR

The T0TR register holds the Memory tags to be used for next memory write transaction. Tags are either individually recorded during memory read transactions performed with address araddr[6:4] or all of them are overridden when software writes to this register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD30

Type

RW

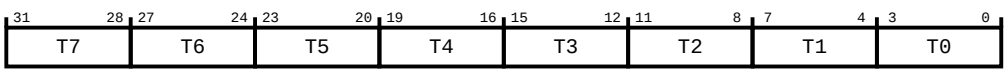
Reset value

0x00000000

Bit descriptions

The following figure shows the T0TR register bit assignments.

Figure 9-320: Bit assignment diagram for the T0TR register



The following table shows the T0TR register bit descriptions.

Table 9-332: T0TR bit descriptions

Bits	Name	Reset	Type	Description
31:28	T7	0b0000	RW	Allocation tag value recorded during latest system memory read transaction performed with address araddr[6:4]=7. Allocation tag value for next system memory write transaction to be performed with address awaddr[6:4]=7.
27:24	T6	0b0000	RW	Allocation tag value recorded during latest system memory read transaction performed with address araddr[6:4]=6. Allocation tag value for next system memory write transaction to be performed with address awaddr[6:4]=6.
23:20	T5	0b0000	RW	Allocation tag value recorded during latest system memory read transaction performed with address araddr[6:4]=5. Allocation tag value for next system memory write transaction to be performed with address awaddr[6:4]=5.
19:16	T4	0b0000	RW	Allocation tag value recorded during latest system memory read transaction performed with address araddr[6:4]=4. Allocation tag value for next system memory write transaction to be performed with address awaddr[6:4]=4.
15:12	T3	0b0000	RW	Allocation tag value recorded during latest system memory read transaction performed with address araddr[6:4]=3. Allocation tag value for next system memory write transaction to be performed with address awaddr[6:4]=3.
11:8	T2	0b0000	RW	Allocation tag value recorded during latest system memory read transaction performed with address araddr[6:4]=2. Allocation tag value for next system memory write transaction to be performed with address awaddr[6:4]=2.
7:4	T1	0b0000	RW	Allocation tag value recorded during latest system memory read transaction performed with address araddr[6:4]=1. Allocation tag value for next system memory write transaction to be performed with address awaddr[6:4]=1.
3:0	T0	0b0000	RW	Allocation tag value recorded during latest system memory read transaction performed with address araddr[6:4]=0. Allocation tag value for next system memory write transaction to be performed with address awaddr[6:4]=0.

9.13.15 css600_axiap_mte Memory Encryption Contexts Register, MECID

The MECID register holds the memory encryption context ID to be used for the next memory transaction.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDDC

Type

RW

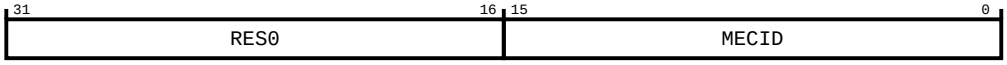
Reset value

0x00000000

Bit descriptions

The following figure shows the MECID register bit assignments.

Figure 9-321: Bit assignment diagram for the MECID register



The following table shows the MECID register bit descriptions.

Table 9-333: MECID bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
15:0	MECID	0x0	RW	MECID value to output on the AXI manager interface.

9.13.16 css600_axiap_mte Configuration Register 1, CFG1

The CFG1 register returns details of the MTE support implemented in the Mem-AP.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDE0

Type

RO

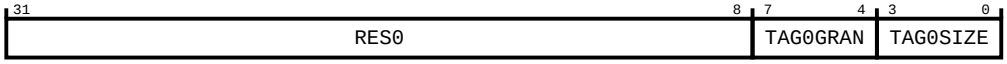
Reset value

0x00000044

Bit descriptions

The following figure shows the CFG1 register bit assignments.

Figure 9-322: Bit assignment diagram for the CFG1 register



The following table shows the CFG1 register bit descriptions.

Table 9-334: CFG1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	TAG0GRAN	0b0100	RO	Memory tagging granule. **0b0100** Memory tag is for 16 data bytes.
3:0	TAG0SIZE	0b0100	RO	Memory tagging support. **0b0100** Memory tagging implemented with 4 tag bits.

9.13.17 css600_axiap_mte Debug Base Address register upper 32 bits, BASEH

The BASEH register holds the upper 32 bits of the base address of a ROM table when LAE is supported.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDF0

Type

RW

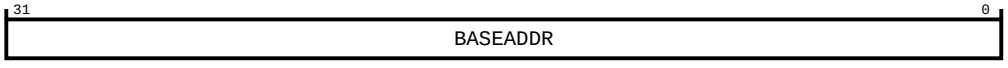
Reset value

0x--------

Bit descriptions

The following figure shows the BASEH register bit assignments.

Figure 9-323: Bit assignment diagram for the BASEH register



The following table shows the BASEH register bit descriptions.

Table 9-335: BASEH bit descriptions

Bits	Name	Reset	Type	Description
31:0	BASEADDR	IMPLEMENTATION DEFINED	RO	Bits [63:32] of the ROM table base address. This register is present only when LAE is supported, otherwise it is reserved and **RAZ/WI**. Even with LAE supported, this field is valid only if BASE.EntryPresent bit is set to 1, in which case it returns the tie-off value of the input signal baseaddr[63:32]. Otherwise, it reads as `0x0`.

9.13.18 css600_axiap_mte Configuration register, CFG

This is the AXI-AP configuration register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xDF4`

Type

RO

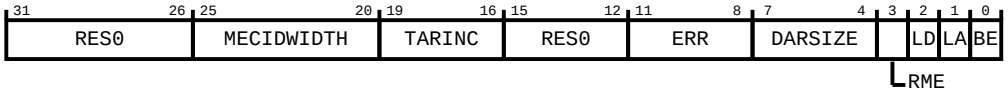
Reset value

`0x010101A-`

Bit descriptions

The following figure shows the CFG register bit assignments.

Figure 9-324: Bit assignment diagram for the CFG register



The following table shows the CFG register bit descriptions.

Table 9-336: CFG bit descriptions

Bits	Name	Reset	Type	Description
31:26	**RES0**	`0b000000`	RO	Reserved bit or field with SBZP behavior.
25:20	MECIDWIDTH	`0b010000`	RO	Memory Encryption Context (MEC) ID width. **`0b010000`** 16-bit MECID width.

Bits	Name	Reset	Type	Description
19:16	TARINC	0b0001	RO	TAR incrementer size. **0b0001** TAR incrementer is 10-bits.
15:12	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
11:8	ERR	0b0001	RO	Identifies the type of error handling that is implemented. **0b0001** TRR, CSW.ERRNPASS and CSW.ERRSTOP are implemented.
7:4	DARSIZE	0b1010	RO	Size of DAR register space. **0b1010** DAR0-DAR255 are implemented.
3	RME	**IMPLEMENTATION DEFINED**	RO	Realm Management Extension **0b0** Realm Management Extension not implemented. **0b1** Realm Management Extension implemented.
2	LD	0b1	RO	Large Data. Indicates support for LDE (data items greater than 32 bits). **0b1** Mem-AP implements the Large Data Extension, and supports data items larger than 32-bits.
1	LA	0b1	RO	Long Address. Indicates support for LAE (greater than 32-bit of addressing). **0b1** Mem-AP supports physical addresses with more than 32 bits. TAR and BASE registers occupy two locations, at 0xD04 and 0xD08, and at 0xDF8 and 0xDF0 respectively.
0	BE	0b0	RO	Big-endian. Always read as 0, BE support is obsolete in Mem-AP from ADIv5.2 **0b0** Not supported.

9.13.19 css600_axiap_mte Debug Base Address register, BASE

Provides an initial system address for the first component in the system. Typically, the system address is the address of a top-level ROM Table that indicates where APv2 APs are located.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDF8

Type

RO

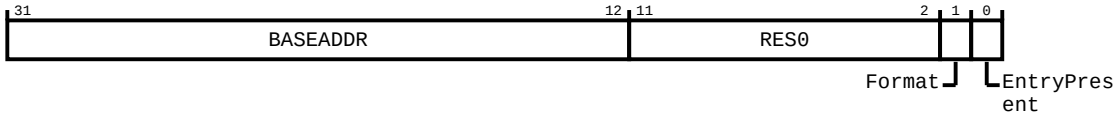
Reset value

`0x-----00-`

Bit descriptions

The following figure shows the BASE register bit assignments.

Figure 9-325: Bit assignment diagram for the BASE register



The following table shows the BASE register bit descriptions.

Table 9-337: BASE bit descriptions

Bits	Name	Reset	Type	Description
31:12	BASEADDR	IMPLEMENTATION DEFINED	RO	Base address of a ROM table. It points to the start of the debug register space or a ROM table address. Bits[11:0] of the address are `0x000` because the address is aligned to 4KB boundary. This field is valid only if BASE.EntryPresent bit is set to 1, in which case it returns the tie-off value of the input signal baseaddr[31:12], otherwise, it reads as `0x0`.
11:2	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
1	Format	`0b1`	RO	Base address register format. Returns the value `0b1` indicating the ADIv5 format, which is unchanged in ADIv6.
0	EntryPresent	IMPLEMENTATION DEFINED	RO	This field indicates whether a debug component is present for this AP. It returns the tie-off value of the input signal baseaddr_valid. **`0b0`** No debug entry present. **`0b1`** Debug entry present and BASE.BASEADDR indicate the start address of the debug register space or ROM table.

9.13.20 css600_axiap_mte Identification Register, IDR

This register provides a mechanism for the debugger to know various identity attributes of the AP.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xDFC`

Type

RO

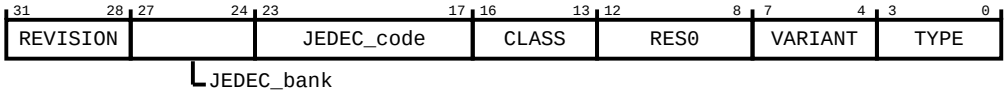
Reset value

0x64770017

Bit descriptions

The following figure shows the IDR register bit assignments.

Figure 9-326: Bit assignment diagram for the IDR register



The following table shows the IDR register bit descriptions.

Table 9-338: IDR bit descriptions

Bits	Name	Reset	Type	Description
31:28	REVISION	0b0110	RO	Revision. An incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
27:24	JEDEC_bank	0b0100	RO	The JEP106 continuation code. **0b0100** Arm
23:17	JEDEC_code	0x3B	RO	The JEP106 identification code. **0x3B** Arm
16:13	CLASS	0b1000	RO	Defines the class of the AP. **0b1000** MEM-AP
12:8	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
7:4	VARIANT	0b0001	RO	Together with the TYPE field, this field identifies the AP implementation. VARIANT differentiates AP implementations that have the same value of TYPE.
3:0	TYPE	0b0111	RO	Indicates the type of bus, or other connection, that connects to the AP. **0b0111** AMBA AXI5 bus.

9.13.21 css600_axiap_mte Integration Test Status register, ITSTATUS

This register indicates the Integration Test DP Abort status.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEFC

Type

RO

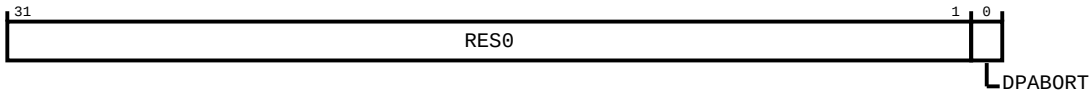
Reset value

0x00000000

Bit descriptions

The following figure shows the ITSTATUS register bit assignments.

Figure 9-327: Bit assignment diagram for the ITSTATUS register



The following table shows the ITSTATUS register bit descriptions.

Table 9-339: ITSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	DPABORT	0b0	RO	When in Integration testing mode (ITCTRL.IME=0b1): Behaves as a sticky bit and latches to 1 on a rising edge of dp_abort. Cleared on a read from this register. If dp_abort rises in the same cycle as a read of the ITSTATUS register is received, the read takes priority and the register is cleared. When in normal functional operation mode (ITCTRL.IME=0b0): Read as 0, writes ignored.

9.13.22 css600_axiap_mte Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

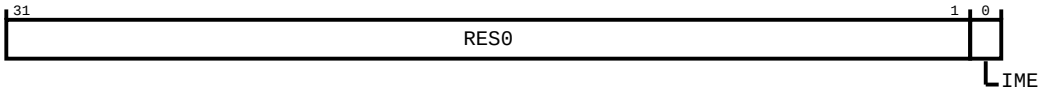
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-328: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-340: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.13.23 css600_axiap_mte Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

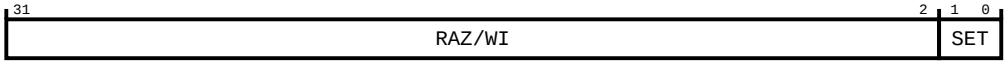
Reset value

0x00000003

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-329: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-341: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RAZ/WI**	`0x0`	RO	**RAZ/WI**.
1:0	SET	`0b11`	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.13.24 css600_axiap_mte Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFA4`

Type

RW

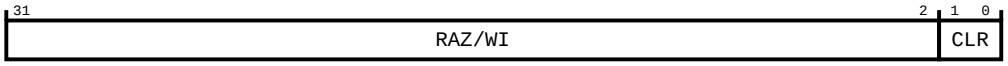
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-330: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-342: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:2	RAZ/WI	0x0	RO	RAZ/WI.
1:0	CLR	0b00	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.13.25 css600_axiap_mte Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

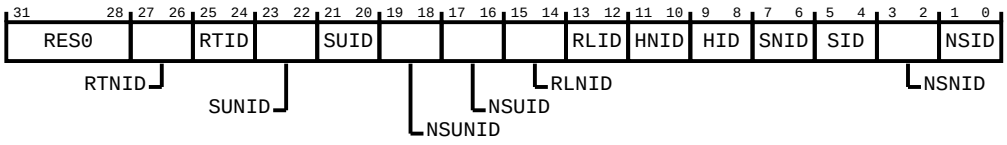
Reset value

0x0-00-0--

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-331: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-343: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	RES0	0b0000	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
27:26	RTNID	**UNKNOWN**	RO	Root non-invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
25:24	RTID	**UNKNOWN**	RO	Root invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	**UNKNOWN**	RO	Realm non-invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
13:12	RLID	**UNKNOWN**	RO	Realm invasive debug. **0b10** Implemented and disabled. **0b11** Implemented and enabled.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
7:6	SNID	**UNKNOWN**	RO	Secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
5:4	SID	**UNKNOWN**	RO	Secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
3:2	NSNID	**UNKNOWN**	RO	Non-secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
1:0	NSID	**UNKNOWN**	RO	Non-secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.

9.13.26 css600_axiap_mte Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFBC`

Type

RO

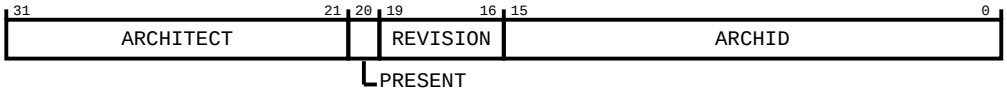
Reset value

`0x47700A17`

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-332: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-344: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x23B	RO	Defines the architect of the component **0x23B** Arm
20	PRESENT	0b1	RO	Indicates the presence of this register **0b1** DEVARCH is present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0A17	RO	Architecture ID. Returns a value that identifies the architecture of the component. **0x0A17** Memory Access Port v2 architecture

9.13.27 css600_axiap_mte Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

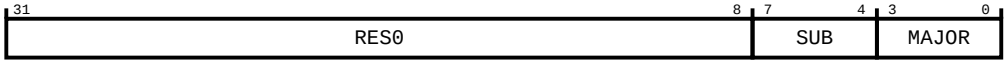
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-333: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-345: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	0b0000	RO	Minor classification. Returns 0x0, Other/undefined.
3:0	MAJOR	0b0000	RO	Major classification. Returns 0x0, Miscellaneous.

9.13.28 css600_axiap_mte Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

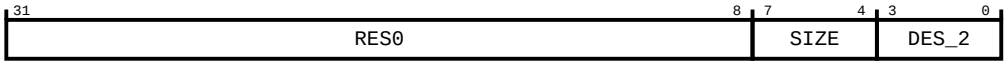
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-334: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-346: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.13.29 css600_axiap_mte Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

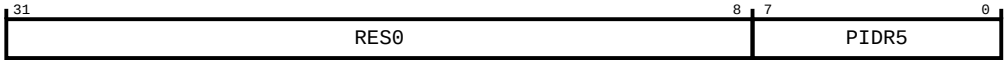
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-335: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-347: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.13.30 css600_axiap_mte Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

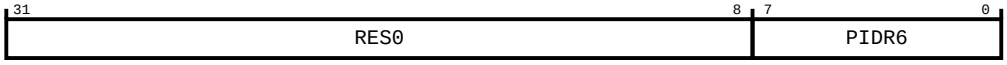
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-336: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-348: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.13.31 css600_axiap_mte Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFDC`

Type

RO

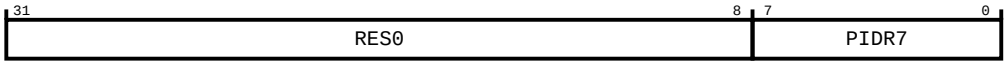
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-337: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-349: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.13.32 css600_axiap_mte Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

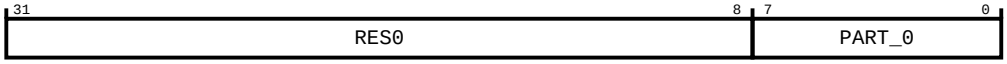
Reset value

0x000000E4

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-338: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-350: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xE4	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.13.33 css600_axiap_mte Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

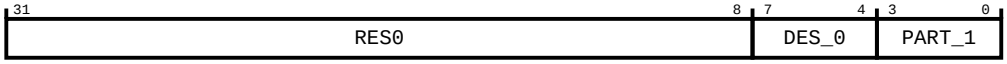
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-339: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-351: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.13.34 css600_axiap_mte Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

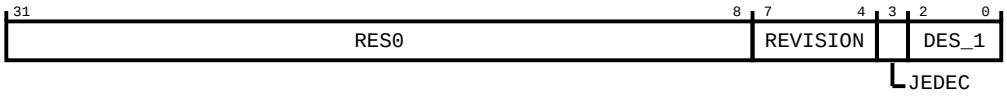
Reset value

0x0000006B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-340: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-352: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b0110	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.13.35 css600_axiap_mte Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

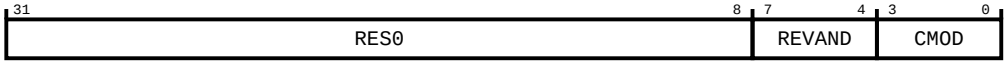
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-341: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-353: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	`0b0000`	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is `0x0`.
3:0	CMOD	`0b0000`	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is `0x0`.

9.13.36 css600_axiap_mte Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

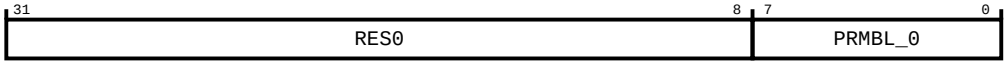
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-342: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-354: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.13.37 css600_axiap_mte Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

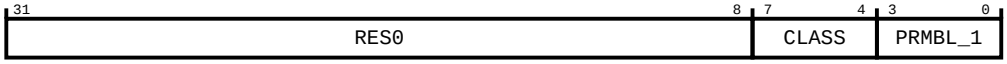
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-343: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-355: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.13.38 css600_axiap_mte Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

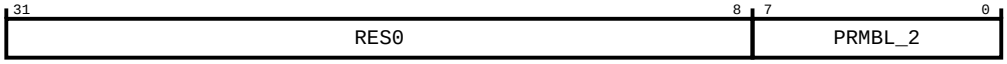
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-344: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-356: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.13.39 css600_axiap_mte Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

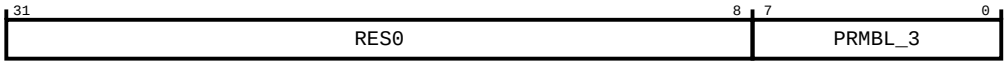
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-345: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-357: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.14 css600_catu register summary

This section describes the css600_catu_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-358: css600_catu_registers register summary

Offset	Name	Type	Reset	Width	Description
0x000	CONTROL	RW	0x00000000	32-bit	CATU control register
0x004	MODE	RW	0x0000000-	32-bit	Mode register
0x008	AXICTRL	RW	0x000000--	32-bit	AXI control register
0x00c	IRQEN	RW	0x0000000-	32-bit	Interrupt enable register
0x020	SLADDRLO	RW	0x-----000	32-bit	Scatter List Address Low register
0x024	SLADDRHI	RW	0x--------	32-bit	Scatter List Address High register
0x028	INADDRLO	RW	0x---00000	32-bit	Input Address Low register
0x02c	INADDRHI	RW	0x--------	32-bit	Input Address High register
0x100	STATUS	RO	0x00000100	32-bit	Status register
0xed0	ITIRQ	WO	0x00000000	32-bit	Integration Test Interrupt register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x0000000F	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x000000--	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x00000000	32-bit	Device Architecture Register
0xfc8	DEVID	RO	0x0000----	32-bit	Device Configuration Register
0xfcc	DEVTYPE	RO	0x00000000	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000EE	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000002B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.14.1 css600_catu CATU control register, CONTROL

The CONTROL register is the global enable register for the CATU. Setting the ENABLE bit enables the CATU. When the CATU is disabled, any received AXI transactions result in an error response. STATUS.READY must be HIGH before the CATU is enabled.

See the ARM AMBA AXI Protocol Specification for information on the bit encodings.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x000`

Type

RW

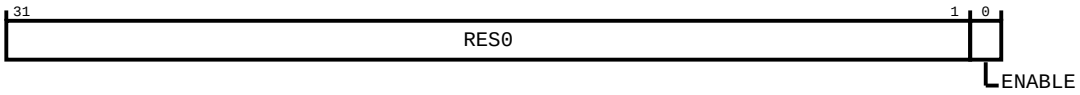
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CONTROL register bit assignments.

Figure 9-346: Bit assignment diagram for the CONTROL register



The following table shows the CONTROL register bit descriptions.

Table 9-359: CONTROL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
0	ENABLE	`0b0`	RW	Enables the CATU: **`0b0`** CATU is disabled **`0b1`** CATU is enabled

9.14.2 css600_catu Mode register, MODE

The MODE register controls the CATU operating mode. There are two modes of operation: Pass-through mode and Translate mode. It is writable only when CONTROL.ENABLE is clear and STATUS.READY is set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x004`

Type

RW

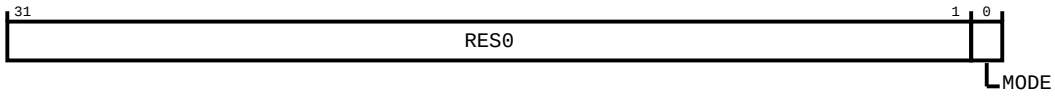
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the MODE register bit assignments.

Figure 9-347: Bit assignment diagram for the MODE register



The following table shows the MODE register bit descriptions.

Table 9-360: MODE bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
0	MODE	**UNKNOWN**	RW	Sets the CATU operating mode: **0b0** Pass-through mode **0b1** Translate mode

9.14.3 css600_catu AXI control register, AXICTRL

The AXICTRL register controls the CATU scatter list read accesses to system memory through the AXI interface. It is writable only when CONTROL.ENABLE is clear and STATUS.READY is set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x008

Type

RW

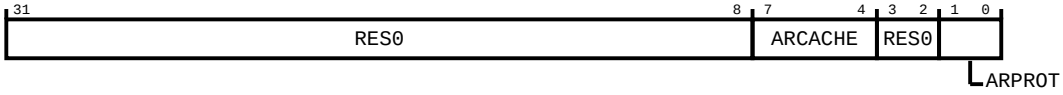
Reset value

0x000000--

Bit descriptions

The following figure shows the AXICTRL register bit assignments.

Figure 9-348: Bit assignment diagram for the AXICTRL register



The following table shows the AXICTRL register bit descriptions.

Table 9-361: AXICTRL bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	ARCACHE	**UNKNOWN**	RW	This field controls the AXI cache encoding, that is, the value that is driven on AXI manager bus arcache_m[3:0], for scatter list read transfers. Software must only program a valid cache encoding value in this field. These values are defined in the AMBA AXI Protocol Specification. Writing a reserved value will set the field to 0b0000.
3:2	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
1:0	ARPROT	**UNKNOWN**	RW	Secure Access, Privileged Access. This field controls the value that is driven on arprot_m[1:0] on the AXI manager interface during scatter list read transfers. The CATU only performs data accesses, so the arprot_m[2] outputs are LOW for all AXI transfers.

9.14.4 css600_catu Interrupt enable register, IRQEN

The IRQEN register is the enable register for the interrupt signal. If the interrupt is enabled, the interrupt signal is asserted when the incoming AXI address transaction is not in the valid address range. It is writable only when CONTROL.ENABLE is clear and STATUS.READY is set.

Disabling the CATU, that is setting the CONTROL.ENABLE register to 0, clears the interrupt signal.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x00C`

Type

RW

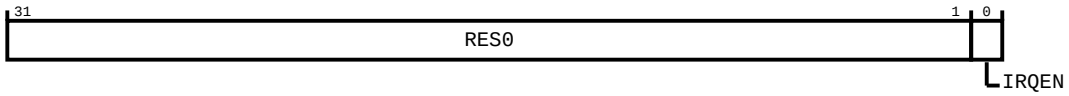
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the IRQEN register bit assignments.

Figure 9-349: Bit assignment diagram for the IRQEN register



The following table shows the IRQEN register bit descriptions.

Table 9-362: IRQEN bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	IRQEN	**UNKNOWN**	RW	Interrupt enable. **0b0** Interrupt is disabled. **0b1** Interrupt is enabled.

9.14.5 css600_catu Scatter List Address Low register, SLADDRLO

The SLADDRLO register, together with the SLADDRHI register, enables the CATU to locate the scatter list in the system memory. Software must program the SLADDRLO register with an initial

value before setting the CONTROL.ENABLE bit to 1. It is writable only when CONTROL.ENABLE is clear and STATUS.READY is set.

The Scatter List in the system memory must be aligned to a 4KB address boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x020

Type

RW

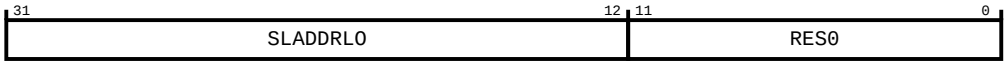
Reset value

0x-----000

Bit descriptions

The following figure shows the SLADDRLO register bit assignments.

Figure 9-350: Bit assignment diagram for the SLADDRLO register



The following table shows the SLADDRLO register bit descriptions.

Table 9-363: SLADDRLO bit descriptions

Bits	Name	Reset	Type	Description
31:12	SLADDRLO	UNKNOWN	RW	Holds bits [31:12] of the lower 32 bits of the AXI address that is used to locate the scatter list in the system memory
11:0	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

9.14.6 css600_catu Scatter List Address High register, SLADDRHI

The SLADDRHI register, together with the SLADDRLO register, enables the CATU to locate the scatter list in the system memory. Software must program the SLADDRHI register with an initial value before setting the CONTROL.ENABLE bit to 1. It is writable only when CONTROL.ENABLE is clear and STATUS.READY is set.

The Scatter List in the system memory must be aligned to a 4KB address boundary.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x024

Type

RW

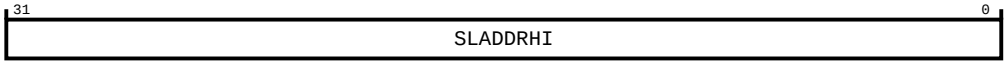
Reset value

0x--------

Bit descriptions

The following figure shows the SLADDRHI register bit assignments.

Figure 9-351: Bit assignment diagram for the SLADDRHI register



The following table shows the SLADDRHI register bit descriptions.

Table 9-364: SLADDRHI bit descriptions

Bits	Name	Reset	Type	Description
31:0	SLADDRHI	UNKNOWN	RW	Holds the upper bits, that is, bit[32] and above, of the AXI address that is used to locate the scatter list in the system memory. RES0 if AXI_ADDR_WIDTH is 32-bits.

9.14.7 css600_catu Input Address Low register, INADDRLO

The INADDRLO register, together with the INADDRHI register, enables the CATU to validate the input address on the AXI subordinate interface. It gives the lower 32 bits of the lower value of the valid input address range.

Software must program the INADDRLO register with an initial value before setting CONTROL.ENABLE bit to 1. It is writable only when CONTROL.ENABLE is clear and STATUS.READY is set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x028

Type

RW

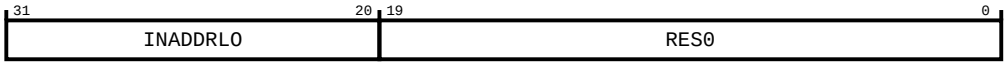
Reset value

0x---00000

Bit descriptions

The following figure shows the INADDRLO register bit assignments.

Figure 9-352: Bit assignment diagram for the INADDRLO register



The following table shows the INADDRLO register bit descriptions.

Table 9-365: INADDRLO bit descriptions

Bits	Name	Reset	Type	Description
31:20	INADDRLO	UNKNOWN	RW	Holds bits [31:20] of the lower 32 bits of the lower value of the valid AXI address range
19:0	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

9.14.8 css600_catu Input Address High register, INADDRHI

The INADDRHI register, together with the INADDRLO register, enables the CATU to validate the input address on the AXI subordinate interface. It gives the upper 32 bits of the lower value of the valid input address range.

Software must program the INADDRHI register with an initial value before setting CONTROL.ENABLE bit to 1. It is writable only when CONTROL.ENABLE is clear and STATUS.READY is set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x02C

Type

RW

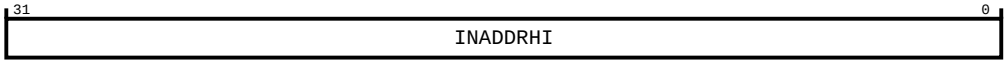
Reset value

0x--------

Bit descriptions

The following figure shows the INADDRHI register bit assignments.

Figure 9-353: Bit assignment diagram for the INADDRHI register



The following table shows the INADDRHI register bit descriptions.

Table 9-366: INADDRHI bit descriptions

Bits	Name	Reset	Type	Description
31:0	INADDRHI	**UNKNOWN**	RW	Holds the upper bits, that is, bit[32] and above, of the lower value of the valid AXI address range. **RES0** if AXI_ADDR_WIDTH is 32-bits.

9.14.9 css600_catu Status register, STATUS

The STATUS register is the status register for the CATU, and the status of any AXI or VA address errors.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x100`

Type

RO

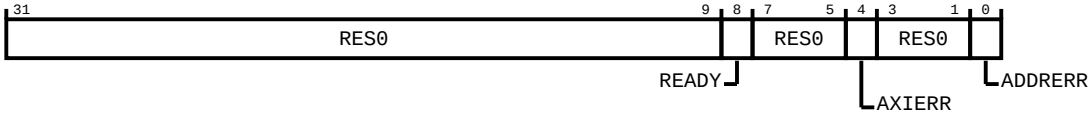
Reset value

`0x00000100`

Bit descriptions

The following figure shows the STATUS register bit assignments.

Figure 9-354: Bit assignment diagram for the STATUS register



The following table shows the STATUS register bit descriptions.

Table 9-367: STATUS bit descriptions

Bits	Name	Reset	Type	Description
31:9	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
8	READY	0b1	RO	CATU ready. This bit indicates that the CATU is inactive and ready for programming. It is set when there are no outstanding AXI transactions: • The AXI interfaces are not busy • The response for final AXI Manager write has been received • There are no outstanding AXI transactions • There are no ongoing translations or translation requests.
7:5	**RES0**	0b000	RO	Reserved bit or field with SBZP behavior.
4	AXIERR	0b0	RO	AXI error. This bit indicates that an error has been detected on the AXI interface. It is set when an error response is received or an unauthenticated transfer is attempted.
3:1	**RES0**	0b000	RO	Reserved bit or field with SBZP behavior.
0	ADDRERR	0b0	RO	Virtual Address error. This bit indicates an error in the incoming VAs. It is set when the VA is less than the address that is defined in the Input Address Low and Input Address High registers, or the VA is larger than sum of the buffer size and the address that is defined in the Input Address Low and Input Address High registers. The buffer size = 4KB * <number of page address entries in the scatter list>.

9.14.10 css600_catu Integration Test Interrupt register, ITIRQ

The ITIRQ register controls the value of the addrerr interrupt output in integration mode, that is when ITCTRL.IME is set. In normal mode this register behaves as **RAZ/WI**. In integration mode the value written to bit[0] of this register is driven on the addrerr output pin.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xED0

Type

WO

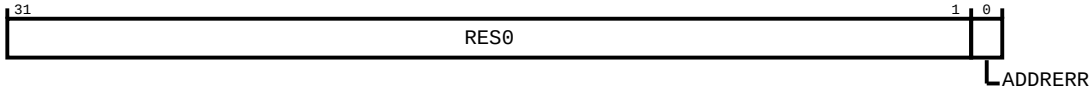
Reset value

0x00000000

Bit descriptions

The following figure shows the ITIRQ register bit assignments.

Figure 9-355: Bit assignment diagram for the ITIRQ register



The following table shows the ITIRQ register bit descriptions.

Table 9-368: ITIRQ bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	ADDRERR	0b0	WO	In Integration Test Mode the addrerr output is directly controlled by this bit.

9.14.11 css600_catu Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

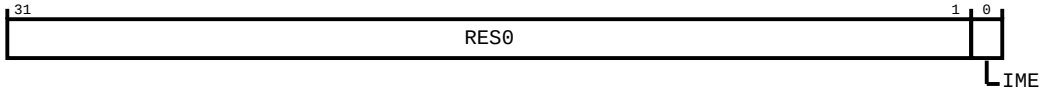
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-356: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-369: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.14.12 css600_catu Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

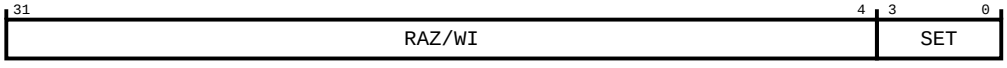
Reset value

0x0000000F

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-357: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-370: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	0x0	RO	**RAZ/WI**.
3:0	SET	0b1111	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.14.13 css600_catu Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFA4`

Type

RW

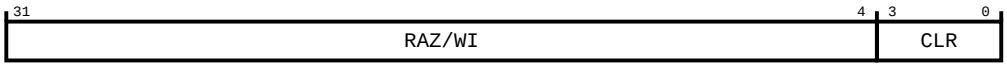
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-358: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-371: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	`0x0`	RO	**RAZ/WI.**
3:0	CLR	`0b0000`	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.14.14 css600_catu Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

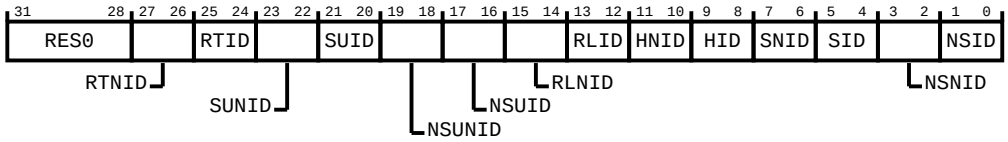
Reset value

0x000000--

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-359: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-372: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	RES0	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	0b00	RO	Root non-invasive debug. **0b00** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.
25:24	RTID	0b00	RO	Root invasive debug. **0b00** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	0b00	RO	Secure non-invasive debug. **0b00** Debug level is not supported.
5:4	SID	UNKNOWN	RO	Secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
3:2	NSNID	0b00	RO	Non-secure non-invasive debug. **0b00** Debug level is not supported.
1:0	NSID	UNKNOWN	RO	Non-secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.

9.14.15 css600_catu Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFBC

Type

RO

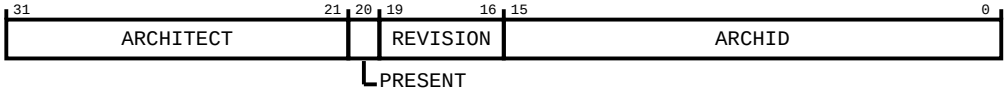
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-360: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-373: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x000	RO	Defines the architect of the component
20	PRESENT	0b0	RO	Indicates the presence of this register **0b0** DEVARCH is not present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0000	RO	Architecture ID. Returns a value that identifies the architecture of the component.

9.14.16 css600_catu Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC8

Type

RO

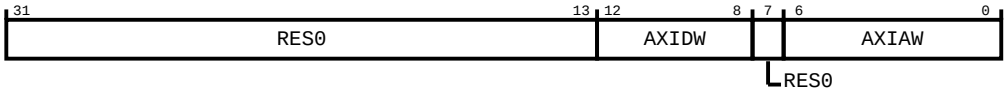
Reset value

0x0000----

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-361: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-374: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:13	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
12:8	AXIDW	IMPLEMENTATION DEFINED	RO	This field indicates the width of the AXI data buses, in multiples of the byte-width. The width of the AXI data buses are set by the parameter AXI_DATA_WIDTH, the default value is 64-bit wide AXI data buses. For example, 0x04 = 32-bit AXI data buses, 0x08 = 64-bit AXI data buses.
7	RES0	0b0	RO	Reserved bit or field with SBZP behavior.
6:0	AXIAW	IMPLEMENTATION DEFINED	RO	This field indicates the width of the AXI address buses, in bits. The width of the AXI address buses are set by the parameter AXI_ADDR_WIDTH, the default value is 40-bit wide AXI address buses.

9.14.17 css600_catu Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

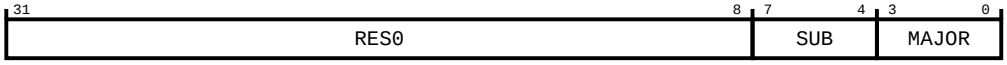
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-362: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-375: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	0b0000	RO	Minor classification. Returns 0x0, Other/undefined.
3:0	MAJOR	0b0000	RO	Major classification. Returns 0x0, Miscellaneous.

9.14.18 css600_catu Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

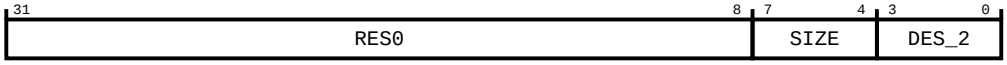
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-363: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-376: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.14.19 css600_catu Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

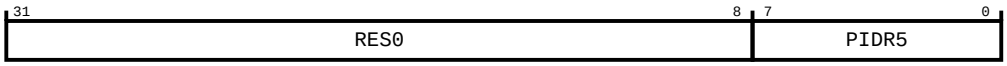
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-364: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-377: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.14.20 css600_catu Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

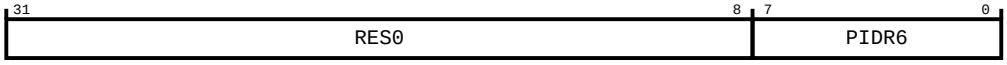
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-365: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-378: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.14.21 css600_catu Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFDC`

Type

RO

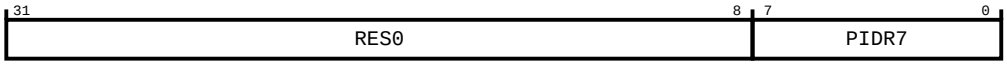
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-366: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-379: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.14.22 css600_catu Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

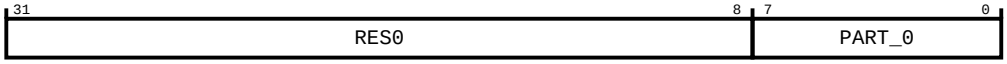
Reset value

0x000000EE

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-367: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-380: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xEE	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.14.23 css600_catu Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

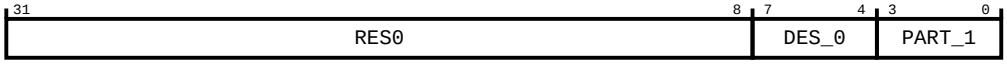
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-368: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-381: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.14.24 css600_catu Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

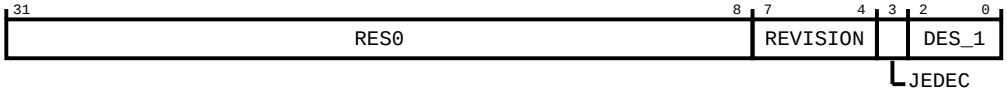
Reset value

0x0000002B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-369: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-382: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b0010	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.14.25 css600_catu Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

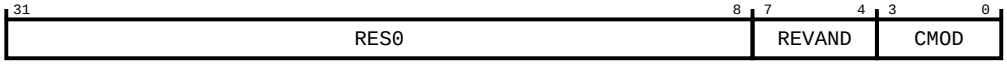
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-370: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-383: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	`0b0000`	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is `0x0`.
3:0	CMOD	`0b0000`	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is `0x0`.

9.14.26 css600_catu Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

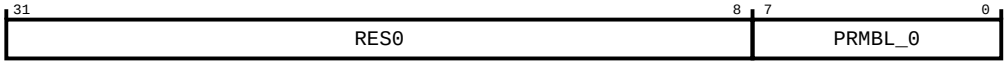
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-371: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-384: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.14.27 css600_catu Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

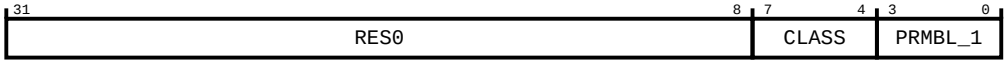
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-372: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-385: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.14.28 css600_catu Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

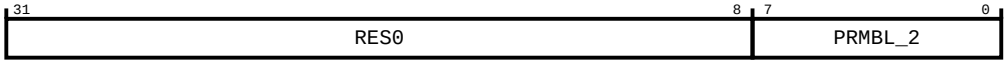
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-373: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-386: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.14.29 css600_catu Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

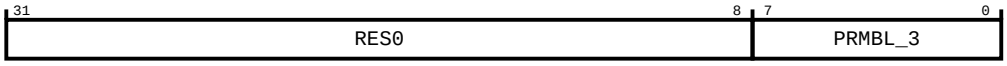
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-374: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-387: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.15 css600_cti register summary

This section describes the css600_cti_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-388: css600_cti_registers register summary

Offset	Name	Type	Reset	Width	Description
0x000	CTICONTROL	RW	0x00000000	32-bit	CTI Control register
0x010	CTIINTACK	WO	0x00000000	32-bit	CTI Interrupt Acknowledge register
0x014	CTIAPPSET	RW	0x00000000	32-bit	CTI Application Channel Set register
0x018	CTIAPPCLEAR	WO	0x00000000	32-bit	CTI Application Channel Clear register
0x01c	CTIAPPPULSE	WO	0x00000000	32-bit	CTI Application Channel Pulse register
0x020	CTIINEN0	RW	0x00000000	32-bit	CTI Trigger 0 to Channel Enable register
0x024	CTIINEN1	RW	0x00000000	32-bit	CTI Trigger 1 to Channel Enable register
0x028	CTIINEN2	RW	0x00000000	32-bit	CTI Trigger 2 to Channel Enable register
...	...				
0x0a0	CTIOUTEN0	RW	0x00000000	32-bit	CTI Channel to Trigger 0 Enable register
0x0a4	CTIOUTEN1	RW	0x00000000	32-bit	CTI Channel to Trigger 1 Enable register
0x0a8	CTIOUTEN2	RW	0x00000000	32-bit	CTI Channel to Trigger 2 Enable register
...	...				
0x11c	CTIOUTEN31	RW	0x00000000	32-bit	CTI Channel to Trigger 31 Enable register
0x130	CTITRIGINSTATUS	RO	0x--------	32-bit	CTI Trigger Input Status register
0x134	CTITRIGOUTSTATUS	RO	0x--------	32-bit	CTI Trigger Output Status register
0x138	CTICHINSTATUS	RO	0x0000----	32-bit	CTI Channel Input Status register
0x13c	CTICHOUTSTATUS	RO	0x0000----	32-bit	CTI Channel Output Status register
0x140	CTIGATE	RW	0x0000FFFF	32-bit	Enable CTI Channel Gate register
0x144	ASICCTRL	RW	0x00000000	32-bit	External Multiplexer Control register
0xee4	ITCHOUT	RW	0x00000000	32-bit	Integration Test Channel Output register
0xee8	ITTRIGOUT	RW	0x00000000	32-bit	Integration Test Trigger Output register
0xef4	ITCHIN	RO	0x00000000	32-bit	Integration Test Channel Input register
0xef8	ITTRIGIN	RO	0x00000000	32-bit	Integration Test Trigger Input register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x0000000F	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfa8	DEVAFF0	RO	0x--------	32-bit	Device Affinity register 0
0xfac	DEVAFF1	RO	0x--------	32-bit	Device Affinity register 1
0xfb8	AUTHSTATUS	RO	0x0000000-	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x47701A14	32-bit	Device Architecture Register
0xfc8	DEVID	RO	0x01------	32-bit	Device Configuration Register
0xfcc	DEVTYPE	RO	0x00000014	32-bit	Device Type Identifier Register

Offset	Name	Type	Reset	Width	Description
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000ED	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000004B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.15.1 css600_cti CTI Control register, CTICONTROL

The CTICONTROL register enables and disables the CTI.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x000

Type

RW

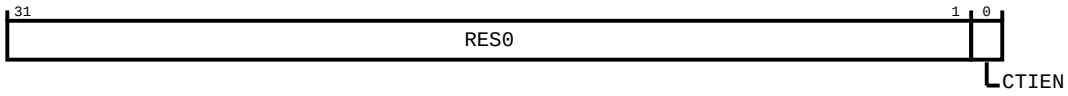
Reset value

0x00000000

Bit descriptions

The following figure shows the CTICONTROL register bit assignments.

Figure 9-375: Bit assignment diagram for the CTICONTROL register



The following table shows the CTICONTROL register bit descriptions.

Table 9-389: CTICONTROL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	CTIEN	0b0	RW	Enable control: **0b0** CTI disabled **0b1** CTI enabled

9.15.2 css600_cti CTI Interrupt Acknowledge register, CTIINTACK

The CTIINTACK register acknowledges trigger outputs. It is a bit map that allows selective clearing of trigger output events. .

If the SW_HANDSHAKE parameter for a trigger output is set, indicating that the output latches HIGH when an event is sent to that output, then the output remains HIGH until the corresponding INTACK bit is written to a 1. A write of a bit to 0 has no effect. This allows different software threads to be responsible for clearing different trigger outputs without needing to perform a read-modify-write operation to find which bits are set..

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x010

Type

WO

Reset value

0x00000000

Bit descriptions

The following figure shows the CTIINTACK register bit assignments.

Figure 9-376: Bit assignment diagram for the CTIINTACK register

The following table shows the CTIINTACK register bit descriptions.

Table 9-390: CTIINTACK bit descriptions

Bits	Name	Reset	Type	Description
31:0	INTACK	0x0	WO	Acknowledges the corresponding event_out output

9.15.3 css600_cti CTI Application Channel Set register, CTIAPPSET

The CTIAPPSET register allows software to set any channel output. Software can use this register to generate a channel event in place of a hardware source on a trigger input. This register must not be used in a system where all events are sent as single-cycle pulses.

It is only retained for compatibility with older systems and software. The register is implemented before the CTIGATE register. Therefore, for the channel event to propagate outside the CTI, the corresponding CTIGATE bit must be set to 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x014

Type

RW

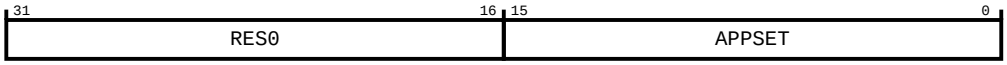
Reset value

0x00000000

Bit descriptions

The following figure shows the CTIAPPSET register bit assignments.

Figure 9-377: Bit assignment diagram for the CTIAPPSET register



The following table shows the CTIAPPSET register bit descriptions.

Table 9-391: CTIAPPSET bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15:0	APPSET	0x0	RW	Sets the corresponding internal channel flag: **0x0** Read: application channel is inactive. Write: has no effect. **0x1** Read: application channel is active. Write: sets the channel output.

9.15.4 css600_cti CTI Application Channel Clear register, CTIAPPCLEAR

The CTIAPPCLEAR register allows software to clear any channel output. Software can use this register to clear a channel event in place of a hardware source on a trigger input. This register must not be used in a system where all events are sent as single-cycle pulses.

It is only retained for compatibility with older systems and software. The register is implemented before the CTIGATE register. Therefore, for the channel event to propagate outside the CTI, the corresponding CTIGATE bit must be set to 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x018

Type

WO

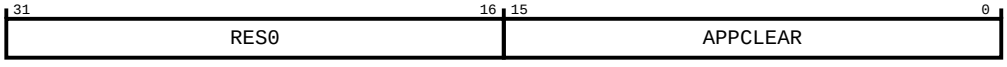
Reset value

0x00000000

Bit descriptions

The following figure shows the CTIAPPCLEAR register bit assignments.

Figure 9-378: Bit assignment diagram for the CTIAPPCLEAR register



The following table shows the CTIAPPCLEAR register bit descriptions.

Table 9-392: CTIAPPCLEAR bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15:0	APPCLEAR	0x0	WO	Clears the corresponding internal channel flag. **0x0** No effect. **0x1** Clears the channel output.

9.15.5 css600_cti CTI Application Channel Pulse register, CTIAPPPULSE

The application channel pulse register allows software to pulse any channel output. This register can be used by software to pulse a channel event in place of a hardware source on a trigger input.

The register is implemented before the CTIGATE register so, for the channel event to propagate outside the CTI, it is necessary for the corresponding CTIGATE bit to be 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x01C

Type

WO

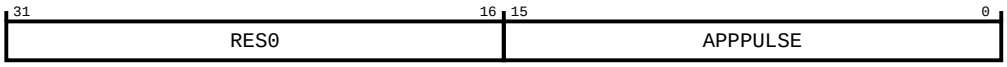
Reset value

0x00000000

Bit descriptions

The following figure shows the CTIAPPPULSE register bit assignments.

Figure 9-379: Bit assignment diagram for the CTIAPPPULSE register



The following table shows the CTIAPPPULSE register bit descriptions.

Table 9-393: CTIAPPPULSE bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15:0	APPPULSE	0x0	WO	Pulses the channel outputs. **0x0** No effect. **0x1** Pulse channel event for one clk cycle.

9.15.6 css600_cti CTI Trigger 0 to Channel Enable register, CTIINEN0

This register maps trigger inputs to channels in the cross trigger system. The CTIINEN registers are bit maps that allow the trigger input to be mapped to any channel output, including none (0x0) and all (2^CHANNEL_WIDTH - 1).

There is one register per trigger input, so it is possible to map any combination of trigger inputs to any channel outputs.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x020

Type

RW

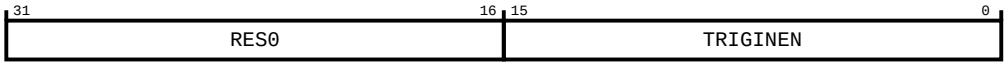
Reset value

0x00000000

Bit descriptions

The following figure shows the CTIINEN0 register bit assignments.

Figure 9-380: Bit assignment diagram for the CTIINEN0 register



The following table shows the CTIINEN0 register bit descriptions.

Table 9-394: CTIINEN0 bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15:0	TRIGINEN	0x0	RW	Trigger input to channel mapping. **0x0** Input trigger 0 events are ignored by the corresponding channel. **0x1** When an event is received on event_in[0], generate an event on the channel corresponding to this bit.

9.15.7 css600_cti CTI Trigger 1 to Channel Enable register, CTIINEN1

This register maps trigger inputs to channels in the cross trigger system. The CTIINEN registers are bit maps that allow the trigger input to be mapped to any channel output, including none (0x0) and all (2^CHANNEL_WIDTH - 1).

There is one register per trigger input, so it is possible to map any combination of trigger inputs to any channel outputs.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x024

Type

RW

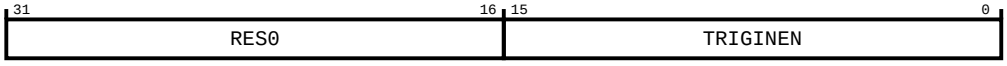
Reset value

0x00000000

Bit descriptions

The following figure shows the CTIINEN1 register bit assignments.

Figure 9-381: Bit assignment diagram for the CTIINEN1 register



The following table shows the CTIINEN1 register bit descriptions.

Table 9-395: CTIINEN1 bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15:0	TRIGINEN	0x0	RW	Trigger input to channel mapping. **0x0** Input trigger 1 events are ignored by the corresponding channel. **0x1** When an event is received on event_in[1], generate an event on the channel corresponding to this bit.

9.15.8 css600_cti CTI Trigger 2 to Channel Enable register, CTIINEN2

This register maps trigger inputs to channels in the cross trigger system. The CTIINEN registers are bit maps that allow the trigger input to be mapped to any channel output, including none (0x0) and all (2^CHANNEL_WIDTH - 1).

There is one register per trigger input, so it is possible to map any combination of trigger inputs to any channel outputs.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x028

Type

RW

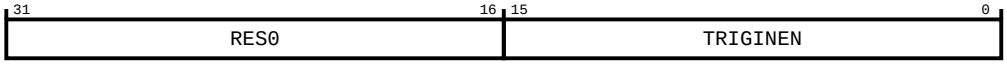
Reset value

0x00000000

Bit descriptions

The following figure shows the CTIINEN2 register bit assignments.

Figure 9-382: Bit assignment diagram for the CTIINEN2 register



The following table shows the CTIINEN2 register bit descriptions.

Table 9-396: CTIINEN2 bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15:0	TRIGINEN	0x0	RW	Trigger input to channel mapping. **0x0** Input trigger 2 events are ignored by the corresponding channel. **0x1** When an event is received on event_in[2], generate an event on the channel corresponding to this bit.

9.15.9 css600_cti CTI Channel to Trigger 0 Enable register, CTIOUTEN0

This register maps channels in the cross trigger system to trigger outputs. The CTIOUTEN registers are bit maps that allow any channel input to be mapped to the trigger output, including none (0x0) and all (2^CHANNEL_WIDTH - 1). .

There is one register per trigger output so it is possible to map any channel input to any combination of trigger outputs..

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x0A0

Type

RW

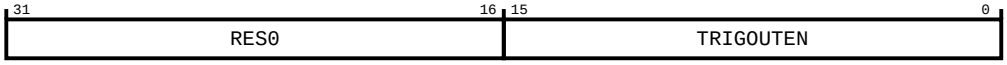
Reset value

0x00000000

Bit descriptions

The following figure shows the CTIOUTEN0 register bit assignments.

Figure 9-383: Bit assignment diagram for the CTIOUTEN0 register



The following table shows the CTIOUTEN0 register bit descriptions.

Table 9-397: CTIOUTEN0 bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15:0	TRIGOUTEN	0x0	RW	Channel to trigger output mapping. **0x0** The corresponding channel is ignored by the output trigger0. **0x1** When an event occurs on the channel corresponding to this bit, generate an event on event_out[0].

9.15.10 css600_cti CTI Channel to Trigger 1 Enable register, CTIOUTEN1

This register maps channels in the cross trigger system to trigger outputs. The CTIOUTEN registers are bit maps that allow any channel input to be mapped to the trigger output, including none (0x0) and all (2^CHANNEL_WIDTH - 1). .

There is one register per trigger output so it is possible to map any channel input to any combination of trigger outputs..

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x0A4

Type

RW

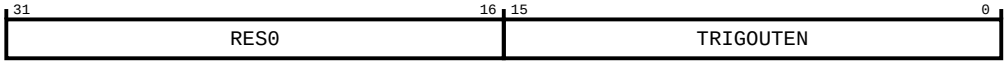
Reset value

0x00000000

Bit descriptions

The following figure shows the CTIOUTEN1 register bit assignments.

Figure 9-384: Bit assignment diagram for the CTIOUTEN1 register



The following table shows the CTIOUTEN1 register bit descriptions.

Table 9-398: CTIOUTEN1 bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15:0	TRIGOUTEN	0x0	RW	Channel to trigger output mapping. **0x0** The corresponding channel is ignored by the output trigger1. **0x1** When an event occurs on the channel corresponding to this bit, generate an event on event_out[1].

9.15.11 css600_cti CTI Channel to Trigger 2 Enable register, CTIOUTEN2

This register maps channels in the cross trigger system to trigger outputs. The CTIOUTEN registers are bit maps that allow any channel input to be mapped to the trigger output, including none (0x0) and all (2^CHANNEL_WIDTH - 1). .

There is one register per trigger output so it is possible to map any channel input to any combination of trigger outputs..

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x0A8

Type

RW

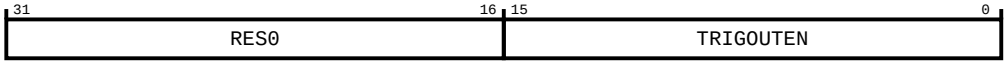
Reset value

0x00000000

Bit descriptions

The following figure shows the CTIOUTEN2 register bit assignments.

Figure 9-385: Bit assignment diagram for the CTIOUTEN2 register



The following table shows the CTIOUTEN2 register bit descriptions.

Table 9-399: CTIOUTEN2 bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15:0	TRIGOUTEN	0x0	RW	Channel to trigger output mapping. **0x0** The corresponding channel is ignored by the output trigger2. **0x1** When an event occurs on the channel corresponding to this bit, generate an event on event_out[2].

9.15.12 css600_cti CTI Channel to Trigger 31 Enable register, CTIOUTEN31

This register maps channels in the cross trigger system to trigger outputs. The CTIOUTEN registers are bit maps that allow any channel input to be mapped to the trigger output, including none (0x0) and all (2^CHANNEL_WIDTH - 1).

There is one register per trigger output so it is possible to map any channel input to any combination of trigger outputs.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x11C

Type

RW

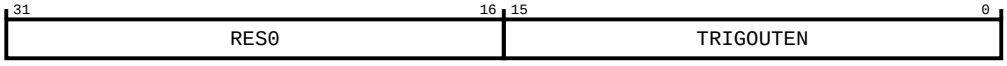
Reset value

0x00000000

Bit descriptions

The following figure shows the CTIOUTEN31 register bit assignments.

Figure 9-386: Bit assignment diagram for the CTIOUTEN31 register



The following table shows the CTIOUTEN31 register bit descriptions.

Table 9-400: CTIOUTEN31 bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15:0	TRIGOUTEN	0x0	RW	Channel to trigger output mapping. **0x0** The corresponding channel is ignored by the output trigger31. **0x1** When an event occurs on the channel corresponding to this bit, generate an event on event_out[31].

9.15.13 css600_cti CTI Trigger Input Status register, CTITRIGINSTATUS

Trigger input status. If the event_in input is driven by a source that generates single cycle pulses, this register is generally read as 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x130

Type

RO

Reset value

0x--------

Bit descriptions

The following figure shows the CTITRIGINSTATUS register bit assignments.

Figure 9-387: Bit assignment diagram for the CTITRIGINSTATUS register

The following table shows the CTITRIGINSTATUS register bit descriptions.

Table 9-401: CTITRIGINSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:0	TRIGINSTATUS	UNKNOWN	RO	Trigger input status. **0x0** One bit per trigger input. 0 means that the input is LOW. **0x1** One bit per trigger input. 1 means that the input is HIGH.

9.15.14 css600_cti CTI Trigger Output Status register, CTITRIGOUTSTATUS

Trigger output status. The register only has meaning if the trigger source drives static levels, rather than pulses.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x134`

Type

RO

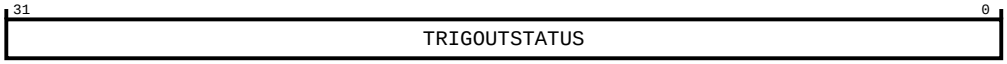
Reset value

`0x--------`

Bit descriptions

The following figure shows the CTITRIGOUTSTATUS register bit assignments.

Figure 9-388: Bit assignment diagram for the CTITRIGOUTSTATUS register



The following table shows the CTITRIGOUTSTATUS register bit descriptions.

Table 9-402: CTITRIGOUTSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:0	TRIGOUTSTATUS	UNKNOWN	RO	Trigger output status. **0x0** One bit per trigger output. 0 means that the output is LOW. **0x1** One bit per trigger output. 1 means that the output is HIGH.

9.15.15 css600_cti CTI Channel Input Status register, CTICHINSTATUS

Channel input status. If the channel input is driven by a source that generates single cycle pulses, this register is generally read as 0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x138`

Type

RO

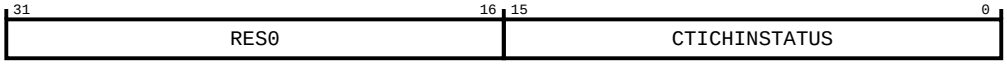
Reset value

`0x0000----`

Bit descriptions

The following figure shows the CTICHINSTATUS register bit assignments.

Figure 9-389: Bit assignment diagram for the CTICHINSTATUS register



The following table shows the CTICHINSTATUS register bit descriptions.

Table 9-403: CTICHINSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:16	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
15:0	CTICHINSTATUS	**UNKNOWN**	RO	Channel input status. **0x0** One bit per channel input. 0 means that the input is LOW. **0x1** One bit per channel input. 1 means that the input is HIGH.

9.15.16 css600_cti CTI Channel Output Status register, CTICHOUTSTATUS

Channel output status. The register only has meaning if the trigger source drives static levels, rather than pulses.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x13C

Type

RO

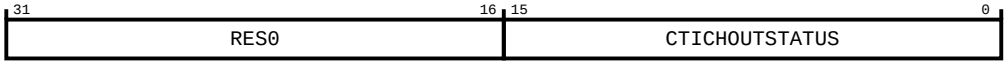
Reset value

0x0000----

Bit descriptions

The following figure shows the CTICHOUTSTATUS register bit assignments.

Figure 9-390: Bit assignment diagram for the CTICHOUTSTATUS register



The following table shows the CTICHOUTSTATUS register bit descriptions.

Table 9-404: CTICHOUTSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:16	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
15:0	CTICHOUTSTATUS	**UNKNOWN**	RO	Channel output status. **0x0** One bit per channel output. 0 means that the output is LOW. **0x1** One bit per channel output. 1 means that the output is HIGH.

9.15.17 css600_cti Enable CTI Channel Gate register, CTIGATE

Channel output gate.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x140

Type

RW

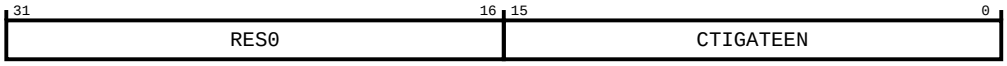
Reset value

0x0000FFFF

Bit descriptions

The following figure shows the CTIGATE register bit assignments.

Figure 9-391: Bit assignment diagram for the CTIGATE register



The following table shows the CTIGATE register bit descriptions.

Table 9-405: CTIGATE bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
15:0	CTIGATEEN	0xFFFF	RW	Enables the propagation of channel events out of the CTI, one bit per channel. **0x0** Disable a channel from propagating. **0x1** Enable channel propagation.

9.15.18 css600_cti External Multiplexer Control register, ASICCTRL

I/O port control.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x144

Type

RW

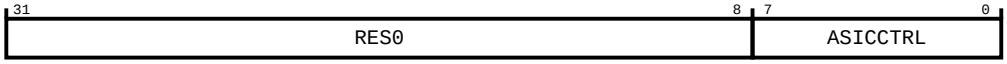
Reset value

0x00000000

Bit descriptions

The following figure shows the ASICCTRL register bit assignments.

Figure 9-392: Bit assignment diagram for the ASICCTRL register



The following table shows the ASICCTRL register bit descriptions.

Table 9-406: ASICCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	ASICCTRL	0x0	RW	Set and clear external output signal. **0x0** Clear output bit to 0. **0x1** Set output bit to 1.

9.15.19 css600_cti Integration Test Channel Output register, ITCHOUT

Integration test mode register, used to generate channel events. Writing to the register creates a single pulse on the output. ITCHOUT is self-clearing.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEE4

Type

RW

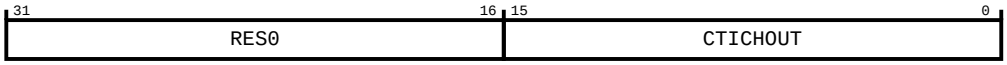
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCHOUT register bit assignments.

Figure 9-393: Bit assignment diagram for the ITCHOUT register



The following table shows the ITCHOUT register bit descriptions.

Table 9-407: ITCHOUT bit descriptions

Bits	Name	Reset	Type	Description
31:16	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
15:0	CTICHOUT	0x0	WO	Pulses the channel outputs. **0x0** No effect. **0x1** Pulse channel event for one clk cycle.

9.15.20 css600_cti Integration Test Trigger Output register, ITTRIGOUT

Integration test mode register, used to generate trigger events.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEE8

Type

RW

Reset value

0x00000000

Bit descriptions

The following figure shows the ITTRIGOUT register bit assignments.

Figure 9-394: Bit assignment diagram for the ITTRIGOUT register

The following table shows the ITTRIGOUT register bit descriptions.

Table 9-408: ITTRIGOUT bit descriptions

Bits	Name	Reset	Type	Description
31:0	CTITRIGOUT	0x0	RW	Set/clear trigger output signal. Reads return the value in the register if SW_HANDSHAKE=1, otherwise 0 is returned if SW_HANDSHAKE=0. Writes: **0x0** Clears the trigger output if SW_HANDSHAKE=1, no effect if SW_HANDSHAKE=0. **0x1** Sets the trigger output if SW_HANDSHAKE=1, pulses trigger output if SW_HANDSHAKE=0.

9.15.21 css600_cti Integration Test Channel Input register, ITCHIN

Integration test mode register, used to view channel events. The integration test register includes a latch that is set when a pulse is received on a channel input. When read, a register bit reads as 1 if the channel has received a pulse since it was last read.

The act of reading the register automatically clears the 1 to a 0. When performing integration testing it is therefore important to coordinate the setting of event latches and reading/clearing them.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF4

Type

RO

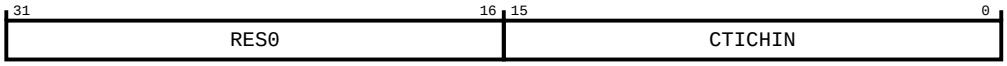
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCHIN register bit assignments.

Figure 9-395: Bit assignment diagram for the ITCHIN register



The following table shows the ITCHIN register bit descriptions.

Table 9-409: ITCHIN bit descriptions

Bits	Name	Reset	Type	Description
31:16	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
15:0	CTICHIN	0x0	RO	Reads the latched value of the channel inputs.

9.15.22 css600_cti Integration Test Trigger Input register, ITTRIGIN

Integration test mode register, used to view trigger events. The integration test register includes a latch that is set when a pulse is received on a trigger input. When read, a register bit reads as 1 if the trigger input has received a pulse since it was last read.

The act of reading the register automatically clears the 1 to a 0. When performing integration testing it is therefore important to coordinate the setting of event latches and reading/clearing them.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF8

Type

RO

Reset value

0x00000000

Bit descriptions

The following figure shows the ITTRIGIN register bit assignments.

Figure 9-396: Bit assignment diagram for the ITTRIGIN register

The following table shows the ITTRIGIN register bit descriptions.

Table 9-410: ITTRIGIN bit descriptions

Bits	Name	Reset	Type	Description
31:0	CTITRIGIN	0x0	RO	Reads the latched value of the trigger inputs.

9.15.23 css600_cti Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xF00`

Type

RW

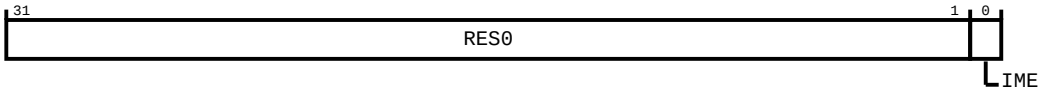
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-397: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-411: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
0	IME	`0b0`	RW	Integration Mode Enable. **`0b0`** The component must enter functional mode. **`0b1`** The component must enter integration mode, and enable support for topology detection and integration testing.

9.15.24 css600_cti Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFA0`

Type

RW

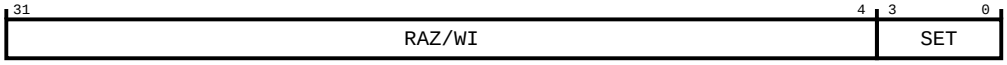
Reset value

`0x0000000F`

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-398: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-412: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	`0x0`	RO	**RAZ/WI**.
3:0	SET	`0b1111`	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.15.25 css600_cti Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA4

Type

RW

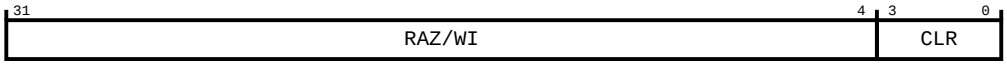
Reset value

0x00000000

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-399: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-413: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	0x0	RO	**RAZ/WI**.
3:0	CLR	0b0000	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.15.26 css600_cti Device Affinity register 0, DEVAFF0

Enables a debugger to determine if two components have an affinity with each other.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA8

Type

RO

Reset value

0x--------

Bit descriptions

The following figure shows the DEVAFF0 register bit assignments.

Figure 9-400: Bit assignment diagram for the DEVAFF0 register

The following table shows the DEVAFF0 register bit descriptions.

Table 9-414: DEVAFF0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	DEVAFF0	IMPLEMENTATION DEFINED	RO	Lower 32-bits of DEVAFF. The value is set by the devaff[31:0] tie-off inputs.

9.15.27 css600_cti Device Affinity register 1, DEVAFF1

Enables a debugger to determine if two components have an affinity with each other.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFAC

Type

RO

Reset value

0x--------

Bit descriptions

The following figure shows the DEVAFF1 register bit assignments.

Figure 9-401: Bit assignment diagram for the DEVAFF1 register

The following table shows the DEVAFF1 register bit descriptions.

Table 9-415: DEVAFF1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	DEVAFF1	IMPLEMENTATION DEFINED	RO	Upper 32-bits of DEVAFF. The value is set by the devaff[63:32] tie-off inputs.

9.15.28 css600_cti Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFB8`

Type

RO

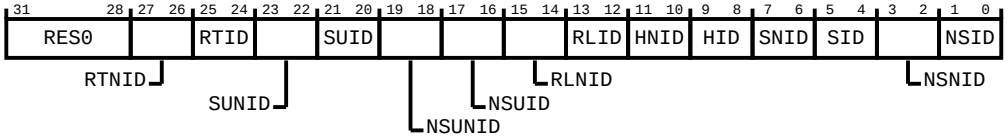
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-402: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-416: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	RES0	`0b0000`	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	`0b00`	RO	Root non-invasive debug. **`0b00`** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.
25:24	RTID	`0b00`	RO	Root invasive debug. **`0b00`** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.

Bits	Name	Reset	Type	Description
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	0b00	RO	Secure non-invasive debug. **0b00** Debug level is not supported.
5:4	SID	0b00	RO	Secure invasive debug. **0b00** Debug level is not supported.
3:2	NSNID	**UNKNOWN**	RO	Non-secure non-invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
1:0	NSID	**UNKNOWN**	RO	Non-secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.

9.15.29 css600_cti Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFBC`

Type

RO

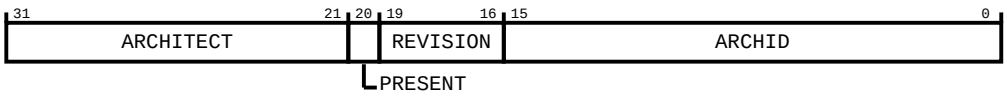
Reset value

`0x47701A14`

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-403: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-417: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	`0x23B`	RO	Defines the architect of the component **`0x23B`** Arm
20	PRESENT	`0b1`	RO	Indicates the presence of this register **`0b1`** DEVARCH is present
19:16	REVISION	`0b0000`	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	`0x1A14`	RO	Architecture ID. Returns a value that identifies the architecture of the component. **`0x1A14`** CTI architecture

9.15.30 css600_cti Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFC8`

Type

RO

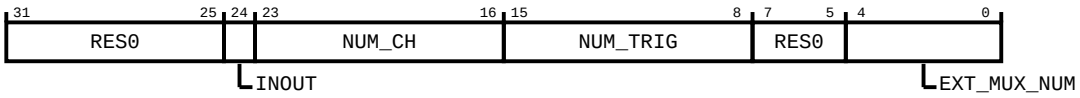
Reset value

`0x01------`

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-404: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-418: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:25	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
24	INOUT	`0b1`	RO	Indicates channel inputs are also masked by the CTIGATE register. Always 1.
23:16	NUM_CH	**IMPLEMENTATION DEFINED**	RO	The number of channels. Always either 4 or 16, set by CHANNEL_WIDTH.
15:8	NUM_TRIG	**IMPLEMENTATION DEFINED**	RO	Indicates the maximum number of triggers - the maximum of the two parameters, NUM_EVENT_RECEIVERS and NUM_EVENT_TRANSMITTERS.
7:5	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
4:0	EXT_MUX_NUM	**IMPLEMENTATION DEFINED**	RO	Indicates the value of the EXT_MUX_NUM parameter, which determines if there is any external multiplexing on the trigger inputs and outputs. 0 indicates no multiplexing.

9.15.31 css600_cti Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFCC`

Type

RO

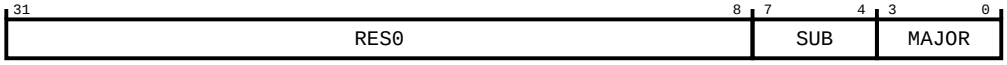
Reset value

`0x00000014`

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-405: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-419: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	`0b0001`	RO	Minor classification. Returns `0x1`, indicating this component is a Trigger-Matrix.
3:0	MAJOR	`0b0100`	RO	Major classification. Returns `0x4`, indicating this component performs Debug Control.

9.15.32 css600_cti Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

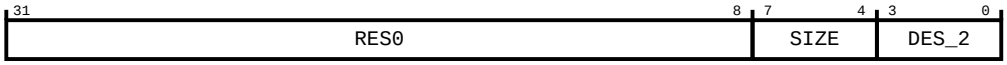
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-406: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-420: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.15.33 css600_cti Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

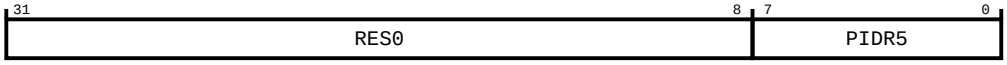
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-407: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-421: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.15.34 css600_cti Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD8

Type

RO

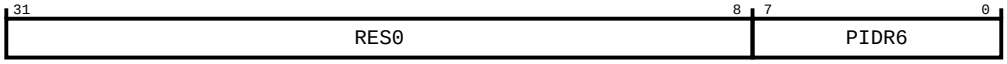
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-408: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-422: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.15.35 css600_cti Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

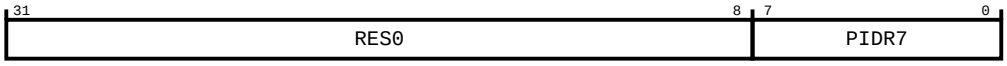
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-409: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-423: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.15.36 css600_cti Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

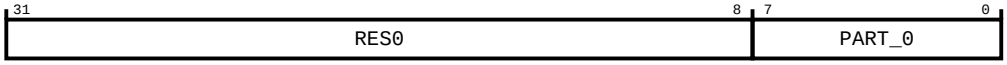
Reset value

0x000000ED

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-410: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-424: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xED	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.15.37 css600_cti Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

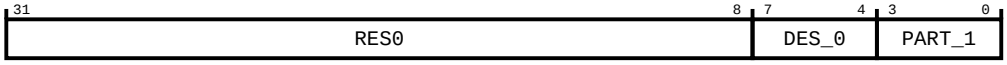
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-411: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-425: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.15.38 css600_cti Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

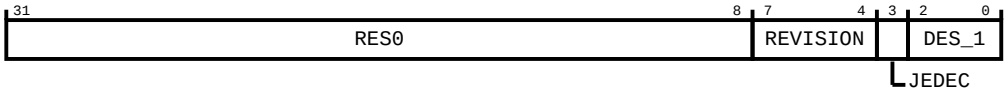
Reset value

0x0000004B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-412: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-426: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b0100	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.15.39 css600_cti Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFEC

Type

RO

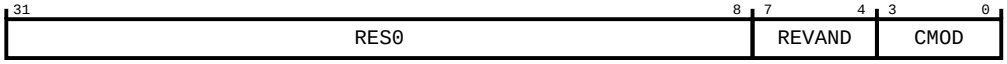
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-413: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-427: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.15.40 css600_cti Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

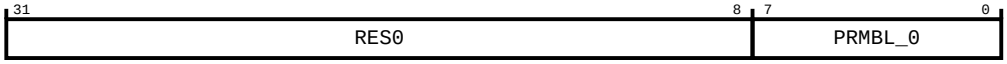
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-414: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-428: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.15.41 css600_cti Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF4`

Type

RO

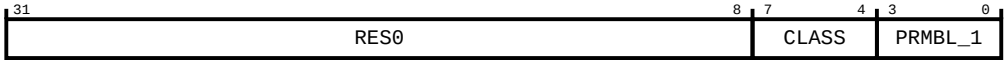
Reset value

`0x00000090`

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-415: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-429: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	`0b1001`	RO	Component class **`0b1001`** CoreSight component
3:0	PRMBL_1	`0b0000`	RO	Preamble. Returns `0x0`.

9.15.42 css600_cti Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

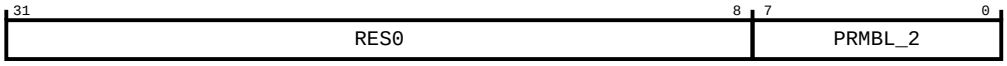
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-416: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-430: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.15.43 css600_cti Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

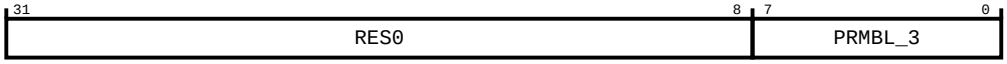
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-417: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-431: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.16 css600_dp register summary

This section describes the css600_dp_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-432: css600_dp_registers register summary

Offset	DPBANKSEL	Name	JTAG-DP	SW-DP	Type	Reset	Width	Description
0x00	-	IDCODE	Yes	No	RO	0x5----477	32-bit	IDCODE Scanchain
0x00	X	ABORT	Yes	Yes	WO	0x00000000	32-bit	Abort Register
0x00	0x00	DPIDR	Yes	Yes	RO	0x5C013477	32-bit	Debug Port Identification Register
0x00	0x01	DPIDR1	Yes	Yes	RO	0x000000--	32-bit	Debug Port Identification Register 1
0x00	0x02	BASEPTR0	Yes	Yes	RO	0x-----00-	32-bit	Base Pointer Register 0
0x00	0x03	BASEPTR1	Yes	Yes	RO	0x00000000	32-bit	Base Pointer Register 1
0x04	0x00	CTRLSTAT	Yes	Yes	RW	0x00000000	32-bit	Control/Status Register
0x04	0x01	DLCR	No	Yes	RW	0x00000040	32-bit	Data Link Control Register
0x04	0x02	TARGETID	Yes	Yes	RO	0x--------	32-bit	Target Identification Register
0x04	0x03	DLPIDR	Yes	Yes	RO	0x-0000001	32-bit	Data Link Protocol Identification Register
0x04	0x04	EVENTSTAT	Yes	Yes	RO	0x0000000-	32-bit	Event Status Register
0x04	0x05	SELECT1	Yes	Yes	WO	0x00000000	32-bit	Select Register 1
0x08	X	RESEND	No	Yes	RO	0x00000000	32-bit	Read Resend Register
0x08	X	SELECT	Yes	Yes	WO	0x00000000	32-bit	Select Register
0x0c	X	RDBUFF			RO	0x00000000	32-bit	Read Buffer Register
0x0c	X	TARGETSEL			WO	0x00000000	32-bit	Target Selection Register

9.16.1 css600_dp IDCODE Scanchain, IDCODE

The IDCODE scanchain identifies the Debug Port JTAG TAP. IDCODE is only present on JTAG-DP.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x00`

Type

RO

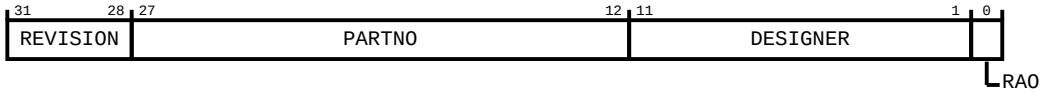
Reset value

`0x5----477`

Bit descriptions

The following figure shows the IDCODE register bit assignments.

Figure 9-418: Bit assignment diagram for the IDCODE register



The following table shows the IDCODE register bit descriptions.

Table 9-433: IDCODE bit descriptions

Bits	Name	Reset	Type	Description
31:28	REVISION	`0b0101`	RO	Revision code.
27:12	PARTNO	**IMPLEMENTATION DEFINED**	RO	Part Number for the DP TAP. **`0xBA06`** DP TAP with 4-bit JTAG Instruction Register. **`0xBA07`** DP TAP with 8-bit JTAG Instruction Register.
11:1	DESIGNER	`0x23B`	RO	Designer ID based on 11-bit JEDEC JEP106 continuation and identity code. `0x23B` - Arm.
0	**RAO**	`0b1`	RO	Read as One.

9.16.2 css600_dp Abort Register, ABORT

The ABORT scanchain or register forces an AP transaction abort. On JTAG-DP ABORT is accessed as a TAP scanchain. On SW-DP ABORT is accessed as a DP register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x00`

Type

WO

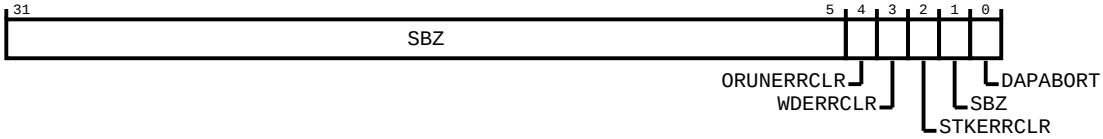
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ABORT register bit assignments.

Figure 9-419: Bit assignment diagram for the ABORT register



The following table shows the ABORT register bit descriptions.

Table 9-434: ABORT bit descriptions

Bits	Name	Reset	Type	Description
31:5	SBZ	`0x0`	RO	Software should write the field as all 0s.
4	ORUNERRCLR	`0b0`	WO	To clear the CTRL/STAT.STICKYORUN overrun error bit to `0b0`, write `0b1` to this bit.
3	WDERRCLR	`0b0`	WO	To clear the CTRL/STAT.WDATAERR write data error bit to `0b0`, write `0b1` to this bit.
2	STKERRCLR	`0b0`	WO	To clear the CTRL/STAT.STICKYERR sticky error bit to `0b0`, write `0b1` to this bit.
1	SBZ	`0b0`	RO	Software should write the field as all 0s.
0	DAPABORT	`0b0`	WO	To generate an AP abort, which aborts the current AP transaction, write `0b1` to this bit. Do this write only if the debugger has received WAIT responses over an extended period.

9.16.3 css600_dp Debug Port Identification Register, DPIDR

The DPIDR provides information about the Debug Port.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x00`

Type

RO

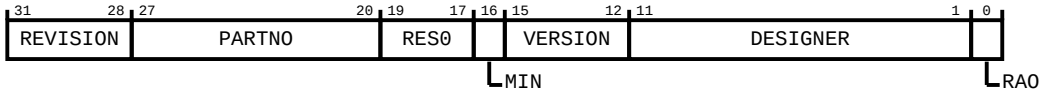
Reset value

`0x5C013477`

Bit descriptions

The following figure shows the DPIDR register bit assignments.

Figure 9-420: Bit assignment diagram for the DPIDR register



The following table shows the DPIDR register bit descriptions.

Table 9-435: DPIDR bit descriptions

Bits	Name	Reset	Type	Description
31:28	REVISION	`0b0101`	RO	Revision code. **`0b0101`** r1p0.
27:20	PARTNO	`0xC0`	RO	Part Number of the Debug Port. `0xC0` - SoC-600.
19:17	**RES0**	`0b000`	RO	Reserved bit or field with SBZP behavior.
16	MIN	`0b1`	RO	Minimal Debug Port: `0b1` - Transaction counter, Pushed-verify, and Pushed-find operations are not implemented.
15:12	VERSION	`0b0011`	RO	Version of Debug Port architecture implemented. `0x3` - DPv3.
11:1	DESIGNER	`0x23B`	RO	Designer ID based on 11-bit JEDEC JEP106 continuation and identity code. `0x23B` - Arm.
0	**RAO**	`0b1`	RO	Reserved, **RAO**.

9.16.4 css600_dp Debug Port Identification Register 1, DPIDR1

The DPIDR1 register is the extension of DPIDR and provides information about the Debug Port.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x01`

Type

RO

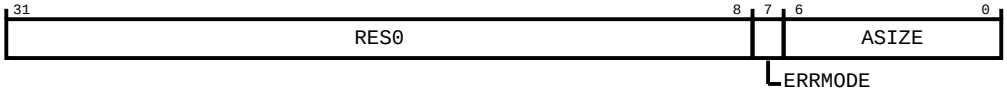
Reset value

`0x000000--`

Bit descriptions

The following figure shows the DPIDR1 register bit assignments.

Figure 9-421: Bit assignment diagram for the DPIDR1 register



The following table shows the DPIDR1 register bit descriptions.

Table 9-436: DPIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7	ERRMODE	`0b1`	RO	Error reporting mode support. **`0b1`** CTRLSTAT.ERRMODE implemented.
6:0	ASIZE	**IMPLEMENTATION DEFINED**	RO	Address size. This field indicates the size of the address in the SELECT, SELECT1, BASEPTR0 and BASEPTR1 registers. **`0xC`** 12-bit address. **`0x14`** 20-bit address. **`0x20`** 32-bit address.

9.16.5 css600_dp Base Pointer Register 0, BASEPTR0

BASEPTR0 and BASEPTR1 provide an initial system address for the first component in the system. Typically, this is the address of a top-level ROM table that indicates where APv2 APs are located.

The size of the address is defined in DPIDR1.ASIZE, which defines the size of the whole address even though bits [11:0] are always zero.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x02`

Type

RO

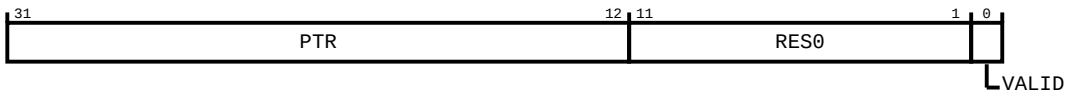
Reset value

`0x-----00-`

Bit descriptions

The following figure shows the BASEPTR0 register bit assignments.

Figure 9-422: Bit assignment diagram for the BASEPTR0 register



The following table shows the BASEPTR0 register bit descriptions.

Table 9-437: BASEPTR0 bit descriptions

Bits	Name	Reset	Type	Description
31:12	PTR	**IMPLEMENTATION DEFINED**	RO	The base address of the first component in the system, formed by concatenating bits[31:0] of BASEPTR1 with bits[31:12] of BASEPTR0. BASEPTR1.PTR provides bits [63:32] of the base address, and BASEPTR0.PTR provides bits [31:12] of the base address. The address is aligned to a 4KB boundary.
11:1	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
0	VALID	**IMPLEMENTATION DEFINED**	RO	Indicates whether the PTR field specifies a valid base address. **`0b0`** No valid base address is specified. The value of the PTR field is **UNKNOWN**. **`0b1`** The PTR field specifies a valid base address.

9.16.6 css600_dp Base Pointer Register 1, BASEPTR1

BASEPTR0 and BASEPTR1 provide an initial system address for the first component in the system. Typically, this is the address of a top-level ROM table that indicates where APv2 APs are located.

The size of the address is defined in DPIDR1.ASIZE, which defines the size of the whole address even though bits [11:0] are always zero.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x03`

Type

RO

Reset value

`0x00000000`

Bit descriptions

The following figure shows the BASEPTR1 register bit assignments.

Figure 9-423: Bit assignment diagram for the BASEPTR1 register

The following table shows the BASEPTR1 register bit descriptions.

Table 9-438: BASEPTR1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	PTR	`0x0`	RO	Base address bits [63:32] of first component in the system. This field is reserved, **RES0**.

9.16.7 css600_dp Control/Status Register, CTRLSTAT

The Control/Status register provides control of the DP and status information about the DP.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x40

Type

RW

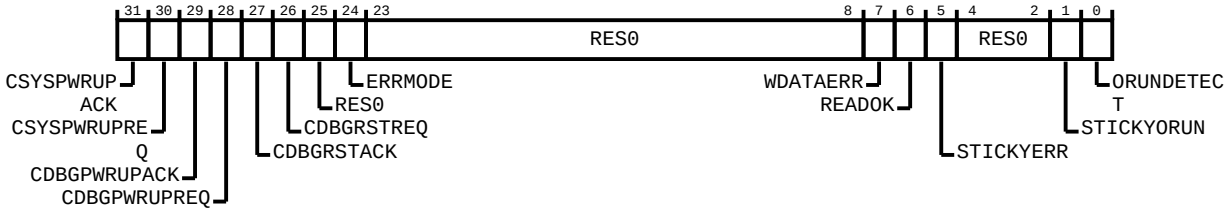
Reset value

0x00000000

Bit descriptions

The following figure shows the CTRLSTAT register bit assignments.

Figure 9-424: Bit assignment diagram for the CTRLSTAT register



The following table shows the CTRLSTAT register bit descriptions.

Table 9-439: CTRLSTAT bit descriptions

Bits	Name	Reset	Type	Description
31	CSYSPWRUPACK	0b0	RO	System powerup acknowledge. Status of CSYSPWRUPACK interface signal.
30	CSYSPWRUPREQ	0b0	RW	System powerup request. This bit controls the CSYSPWRUPREQ interface signal.
29	CDBGPWRUPACK	0b0	RO	Debug powerup acknowledge. Status of CDBGPWRUPACK interface signal.
28	CDBGPWRUPREQ	0b0	RW	Debug powerup request. This bit controls the CDBGPRWUPREQ interface signal.
27	CDBGRSTACK	0b0	RO	Debug reset acknowledge. Indicates the status of the CDBGRSTACK interface signal.
26	CDBGRSTREQ	0b0	RW	Debug reset request. This bit controls the CDBGRSTREQ interface signal.
25	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
24	ERRMODE	0b0	RW	Error Mode. Indicates the reset behavior of the CTRLSTAT.STICKYERR field. ERRMODE can have one of the following values. After a powerup reset, the value of this field is 0b0. **0b0** Errors on AP transactions set CTRLSTAT.STICKYERR and CTRL/STAT.STICKYERR remains set until explicitly cleared. **0b1** Errors on AP transactions set CTRLSTAT.STICKYERR and CTRL/STAT.STICKYERR is cleared when a FAULT response is output.
23:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7	WDATAERR	0b0	RO	This bit is set to 0b1 if one of the following Write Data Error occurs: A parity or framing error on the data phase of a write. A write that has been accepted by the DP is then discarded without being submitted to the AP. Access to and how to clear this field are DATA LINK DEFINED: on JTAG-DP access is reserved, **RES0**. on SW-DP access is RO/**WI**. To clear WDATAERR to 0b0, write 0b1 to the ABORT.WDERRCLR field in the ABORT register. After clearing the WDATAERR flag, you must typically resend the corrupted data. After a powerup reset, WDATAERR is 0b0.

Bits	Name	Reset	Type	Description
6	READOK	0b0	RO	This bit is DATA LINK DEFINED: on JTAG-DP access is reserved, **RES0**. on SW-DP access is RO/**WI**. If the response to the previous AP read or RDBUFF read was OK, the bit is set to 0b1. If the response was not OK, it is cleared to 0b0. This flag always indicates the response to the last AP read access. After a powerup reset, this bit is 0b0.
5	STICKYERR	0b0	RW	If errors are configured to be sticky by CTRLSTAT.ERRMODE, this bit is set to 0b1 when an error is returned by an AP transaction. Access to and how to clear this field are DATA LINK DEFINED: on a JTAG-DP access is R/W1C. To clear STICKYERR to 0b0, write 0b1 to it or write 0b1 to the ABORT.STKERRCLR field. on a SW-DP access is RO/**WI**. To clear STICKYERR to 0b0, write 0b1 to the ABORT.STKERRCLR field. After clearing CTRL/STAT.STICKYERR, you must find the location where the error that caused the flag to be set occurred. After a powerup reset, this bit is 0b0.
4:2	**RES0**	0b000	RO	Reserved bit or field with SBZP behavior.
1	STICKYORUN	0b0	RW	If overrun detection is enabled by CTRLSTAT.ORUNDETECT, this bit is set to 0b1 when an overrun occurs. Access to and how to clear this field are DATA LINK DEFINED: on a JTAG-DP access is R/W1C. To clear STICKYORUN to 0b0, write 0b1 to it or write 0b1 to the ABORT.STKERRCLR field. on a SW-DP access is RO/**WI**. To clear STICKYORUN to 0b0, write 0b1 to the ABORT.ORUNERRCLR field in the ABORT register. After clearing CTRL/STAT.STICKYORUN, you must find out which DP or AP transaction initiated the overrun that caused the flag to be set, and repeat the transactions for that DP or AP from the transaction pointed to by the transaction counter. After a powerup reset, this bit is 0b0.
0	ORUNDETECT	0b0	RW	This bit can have one of the following values. After a powerup reset this bit is 0b0. **0b0** Overrun detection is disabled. **0b1** Overrun detection is enabled.

9.16.8 css600_dp Data Link Control Register, DLCR

The DLCR controls the operating mode of the Data link. Access to this register is DATA LINK DEFINED. For JTAG-DP, this register is Reserved **RES0**.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x41

Type

RW

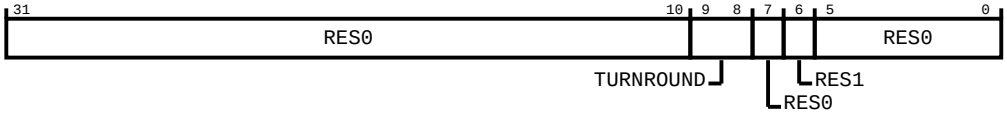
Reset value

0x00000040

Bit descriptions

The following figure shows the DLCR register bit assignments.

Figure 9-425: Bit assignment diagram for the DLCR register



The following table shows the DLCR register bit descriptions.

Table 9-440: DLCR bit descriptions

Bits	Name	Reset	Type	Description
31:10	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
9:8	TURNROUND	0b00	RW	For an SW-DP, this field defines the turnaround tristate period. After a powerup reset, this field is 0b00. **0b00** 1 data period. **0b01** 2 data periods. **0b10** 3 data periods. **0b11** 4 data periods.
7	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
6	**RES1**	0b1	RO	Reserved, **RES1**.
5:0	**RES0**	0b000000	RO	Reserved bit or field with SBZP behavior.

9.16.9 css600_dp Target Identification Register, TARGETID

The TARGETID register provides information about the target when the host is connected to a single device.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x42

Type

RO

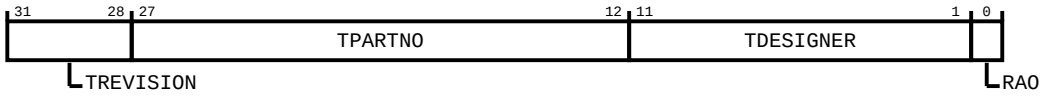
Reset value

0x--------

Bit descriptions

The following figure shows the TARGETID register bit assignments.

Figure 9-426: Bit assignment diagram for the TARGETID register



The following table shows the TARGETID register bit descriptions.

Table 9-441: TARGETID bit descriptions

Bits	Name	Reset	Type	Description
31:28	TREVISION	**IMPLEMENTATION DEFINED**	RO	Target revision. The value comes from the tie-off signal targetid[31:28].
27:12	TPARTNO	**IMPLEMENTATION DEFINED**	RO	Target part number. The value comes from the tie-off signal targetid[27:12].
11:1	TDESIGNER	**IMPLEMENTATION DEFINED**	RO	Designer ID, based on 11-bit JEDEC JEP106 continuation and identity code. The value comes from the tie-off signal targetid[11:1].
0	**RAO**	0b1	RO	Reserved, **RAO**.

9.16.10 css600_dp Data Link Protocol Identification Register, DLPIDR

The DLPIDR provides protocol version information. The contents of this register are DATA LINK DEFINED.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x43

Type

RO

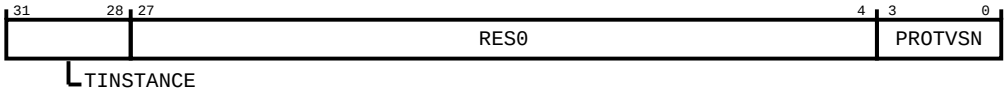
Reset value

0x-0000001

Bit descriptions

The following figure shows the DLPIDR register bit assignments.

Figure 9-427: Bit assignment diagram for the DLPIDR register



The following table shows the DLPIDR register bit descriptions.

Table 9-442: DLPIDR bit descriptions

Bits	Name	Reset	Type	Description
31:28	TINSTANCE	**IMPLEMENTATION DEFINED**	RO	The instance number for this device. For JTAG-DP: Reserved, **RES0**, and for SW-DP: The value comes from the tie-off signal instanceid[3:0]. The value must be unique for all devices with identical TARGETID.TPARTNO and TARGETID.TDESIGNER fields that are connected together in a multi-drop system.
27:4	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
3:0	PROTVSN	`0b0001`	RO	Defines the protocol version that is implemented. For JTAG-DP: `0x1` JTAG-DP protocol version 1. For SW-DP: `0x1` SWD protocol version 2.

9.16.11 css600_dp Event Status Register, EVENTSTAT

The EVENTSTAT register is used by the system to signal an event to the external debugger. EVENTSTAT.EA indicates the inverted value of input signal dp_eventstatus.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x44`

Type

RO

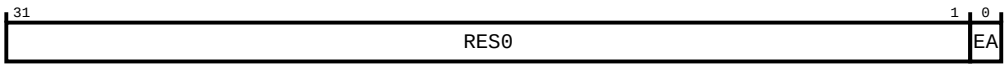
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the EVENTSTAT register bit assignments.

Figure 9-428: Bit assignment diagram for the EVENTSTAT register



The following table shows the EVENTSTAT register bit descriptions.

Table 9-443: EVENTSTAT bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	EA	**UNKNOWN**	RO	Event status flag. Valid values for this bit are: **0b0** An event requires attention. **0b1** There is no event requiring attention.

9.16.12 css600_dp Select Register 1, SELECT1

The SELECT and SELECT1 registers select the DP address bank and the address that is driven on the APB requester interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x45

Type

WO

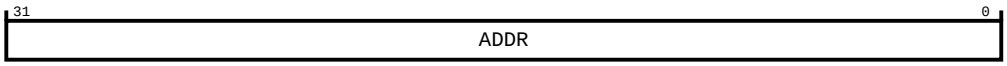
Reset value

0x00000000

Bit descriptions

The following figure shows the SELECT1 register bit assignments.

Figure 9-429: Bit assignment diagram for the SELECT1 register



The following table shows the SELECT1 register bit descriptions.

Table 9-444: SELECT1 bit descriptions

Bits	Name	Reset	Type	Description
31:0	ADDR	0x0	WO	Address output bits[63:32]. This field is ignored because the DP supports a maximum address width of 32-bits.

9.16.13 css600_dp Read Resend Register, RESEND

The Resend Register is DATA LINK DEFINED: JTAG-DP: Reserved. SW-DP: The Resend Register presents data that was returned by the last AP read or DP RDBUFF read.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x80`

Type

RO

Reset value

`0x00000000`

Bit descriptions

The following figure shows the RESEND register bit assignments.

Figure 9-430: Bit assignment diagram for the RESEND register

The following table shows the RESEND register bit descriptions.

Table 9-445: RESEND bit descriptions

Bits	Name	Reset	Type	Description
31:0	RDATA	`0x0`	RO	JTAG-DP: **RES0** SW-DP: Data for previous AP read.

9.16.14 css600_dp Select Register, SELECT

The SELECT and SELECT1 registers select the DP address bank and the address that is driven on the APB requester interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x80`

Type

WO

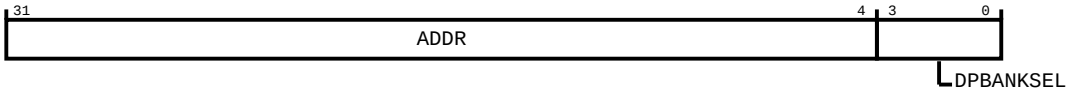
Reset value

0x00000000

Bit descriptions

The following figure shows the SELECT register bit assignments.

Figure 9-431: Bit assignment diagram for the SELECT register



The following table shows the SELECT register bit descriptions.

Table 9-446: SELECT bit descriptions

Bits	Name	Reset	Type	Description
31:4	ADDR	0x0	WO	Address output bits[31:4]. The ADDR field selects a four-word bank of system locations to access. Bits[3:2] of the address, which are used to select a specific register in a bank, are provided with APACC transactions. Bits[1:0] are always 0b00. Note: The size of the address bus is defined in DPIDR1.ASIZE.
3:0	DPBANKSEL	0b0000	WO	Debug Port address bank select.

9.16.15 css600_dp Read Buffer Register, RDBUFF

The purpose and behavior of RDBUFF is DATA LINK DEFINED: JTAG-DP: The Read Buffer is architecturally defined to provide a DP read operation that does not have any side effects.

This definition allows a debugger to insert a DP read of RDBUFF at the end of a sequence of operations, to return the final AP Read Result and ACK values. SW-DP: The Read Buffer presents data that was captured during the previous AP read, enabling returning the value without generating a new AP access. Note: After reading the DP Read Buffer, its contents are no longer valid. The result of a second read of the DP Read Buffer is **UNKNOWN**.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC0

Type

RO

Reset value

0x00000000

Bit descriptions

The following figure shows the RDBUFF register bit assignments.

Figure 9-432: Bit assignment diagram for the RDBUFF register

The following table shows the RDBUFF register bit descriptions.

Table 9-447: RDBUFF bit descriptions

Bits	Name	Reset	Type	Description
31:0	RDATA	0x0	RO	JTAG-DP: **RES0** SW-DP: Data for previous AP read.

9.16.16 css600_dp Target Selection Register, TARGETSEL

The TARGETSEL register selects the target device in a Serial Wire Debug multi-drop system. On a JTAG-DP, any access to this register is reserved, **RES0**.

For SW-DP, n a write to TARGETSEL immediately following a line reset sequence, the target is selected if both the following conditions are met: Bits[31:28] match bits[31:28] in the DLPIDR. Bits[27:0] match bits[27:0] in the TARGETID register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xC0

Type

WO

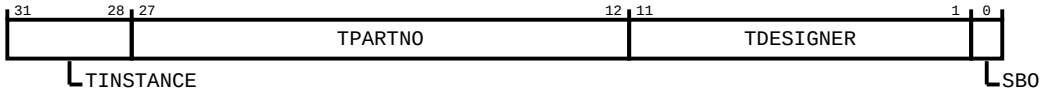
Reset value

0x00000000

Bit descriptions

The following figure shows the TARGETSEL register bit assignments.

Figure 9-433: Bit assignment diagram for the TARGETSEL register



The following table shows the TARGETSEL register bit descriptions.

Table 9-448: TARGETSEL bit descriptions

Bits	Name	Reset	Type	Description
31:28	TINSTANCE	0b0000	WO	The instance number for this device. See DLPIDR.
27:12	TPARTNO	0x0	WO	The value that is assigned by the designer of the part. See TARGETID.
11:1	TDESIGNER	0x0	WO	The 11-bit code that is formed from the JEDEC JEP106 continuation code and identity code. See TARGETID.
0	SBO	0b0	WO	Reserved, SBO.

9.17 css600_jtagap register summary

This section describes the css600_jtagap_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-449: css600_jtagap_registers register summary

Offset	Name	Type	Reset	Width	Description
0xd00	CSW	RW	0x00000000	32-bit	Control/Status Word register
0xd04	PSEL	RW	0x00000000	32-bit	Port Select register
0xd08	PSTA	RW	0x00000000	32-bit	Port Status register
0xd10	BFIFO1	RW	0x000000--	32-bit	Byte FIFO Register 1
0xd14	BFIFO2	RW	0x0000----	32-bit	Byte FIFO Register 2
0xd18	BFIFO3	RW	0x00------	32-bit	Byte FIFO Register 3
0xd1c	BFIFO4	RW	0x--------	32-bit	Byte FIFO Register 4
0xdfc	IDR	RO	0x44760020	32-bit	Identification Register
0xefc	ITSTATUS	RO	0x00000000	32-bit	Integration Test Status register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x00000003	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfbc	DEVARCH	RO	0x47700A27	32-bit	Device Architecture Register
0xfcc	DEVTYPE	RO	0x00000000	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6

Offset	Name	Type	Reset	Width	Description
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000E6	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000004B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.17.1 css600_jtagap Control/Status Word register, CSW

The CSW register configures and controls transfers through the JTAG interface to the connected memory system.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD00

Type

RW

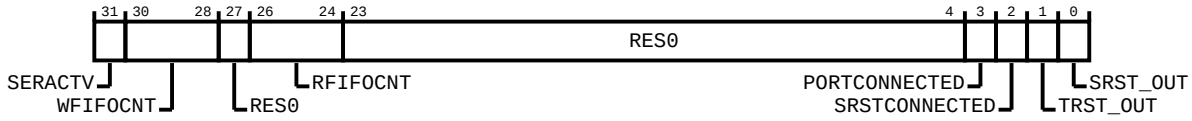
Reset value

0x00000000

Bit descriptions

The following figure shows the CSW register bit assignments.

Figure 9-434: Bit assignment diagram for the CSW register



The following table shows the CSW register bit descriptions.

Table 9-450: CSW bit descriptions

Bits	Name	Reset	Type	Description
31	SERACTV	0b0	RO	JTAG engine active. This bit gets set when the JTAG engine picks the first command from the Command FIFO for execution and remains set until all commands have been executed, that is until after CSW.WFIFOCNT becomes 0 and the JTAG engine goes to idle state. **0b0** JTAG engine is inactive. **0b1** JTAG engine is processing commands from the Command FIFO.
30:28	WFIFOCNT	0b000	RO	Command FIFO outstanding byte count. The reset value is 0x0. Returns the number of command bytes held in the Command FIFO that are yet to be processed by the JTAG engine. Since the Command FIFO is 4 entries deep, this field can only take values between 0 and 4.
27	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
26:24	RFIFOCNT	0b000	RO	Response FIFO outstanding byte count. The reset value is 0x0. Returns the number of bytes of response data held in the Response FIFO. Since the Response FIFO is 7 entries deep, this field can take any value between 0 and 7.
23:4	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
3	PORTCONNECTED	0b0	RO	PORT connected. This bit indicates the logical AND of port_connected inputs from all ports that are currently selected in the PSEL register.
2	SRSTCONNECTED	0b0	RO	SRST connected. This bit is logical AND of srst_connected inputs from all ports that are currently selected in PSEL register.
1	TRST_OUT	0b0	RW	This bit specifies the value to drive out on the active-LOW cs_ntrst pin for the ports that are connected, selected, and their PSTA bit is clear. This bit does not self-clear and must be cleared by a software write to this register. **0b0** De-assert cs_ntrst HIGH. **0b1** Assert cs_ntrst LOW.
0	SRST_OUT	0b0	RW	This bit specifies the value to drive out on the active-LOW srst_out_n pin for the ports that are connected, selected, and their PSTA bit is clear. This bit does not self-clear and must be cleared by a software write to this register. **0b0** De-assert srst_out_n HIGH. **0b1** Assert srst_out_n LOW.

9.17.2 css600_jtagap Port Select register, PSEL

The PSEL register enables JTAG ports if the interface is connected to the JTAG AP and the port_enabled signal from the interface to the JTAG AP is asserted HIGH.

The PSEL register must be written only when the following conditions are met: the JTAG engine is idle and the write FIFO is empty. If this register is written to in any other state, the corresponding JTAG ports are abruptly enabled, or disabled, in the middle of a transfer, which might cause errors, stalls, or deadlocks in the JTAG receiver.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD04`

Type

RW

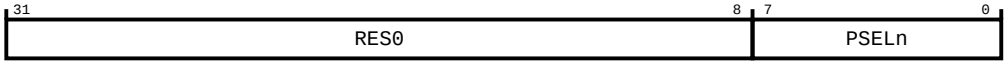
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PSEL register bit assignments.

Figure 9-435: Bit assignment diagram for the PSEL register



The following table shows the PSEL register bit descriptions.

Table 9-451: PSEL bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PSELn	0x0	RW	Port Select. Each register field is named as PSELn, where n = 0-7. The numerical index represents the bit position of that field in this register.

9.17.3 css600_jtagap Port Status register, PSTA

The PSTA register captures the state of a connected and selected port on every clock cycle. If a connected and selected port is disabled or powered down, that is the signal port_enabled goes LOW, even transiently, the corresponding bit in the PSTA register is set in the next cycle.

It remains 1 until it is cleared by writing 1 to it. It gets cleared automatically on abort. Deselecting a port in PSEL does not alter the state of PSTA. If the PSTA bit is set for a port, that port is disabled and its TCK, TMS, and TDI outputs are driven LOW until its PSTA bit is cleared. Software must not clear any PSTA bit unless the JTAG-AP is idle, that is CSW.SERACTV=0b0 and CSW.WFIFOCNT=0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD08

Type

RW

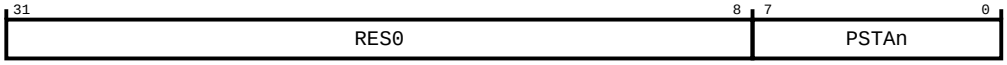
Reset value

0x00000000

Bit descriptions

The following figure shows the PSTA register bit assignments.

Figure 9-436: Bit assignment diagram for the PSTA register



The following table shows the PSTA register bit descriptions.

Table 9-452: PSTA bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PSTAn	0x0	RW	Port Status. Each register field is named as PSTAn, where n = 0-7. The numerical index represents the bit position of that field in this register.

9.17.4 css600_jtagap Byte FIFO Register 1, BFIFO1

The BFIFO1 register enables 1 byte to be transacted with the Response or Command FIFO.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD10

Type

RW

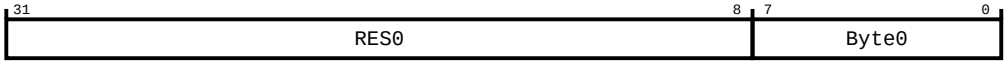
Reset value

0x000000--

Bit descriptions

The following figure shows the BFIFO1 register bit assignments.

Figure 9-437: Bit assignment diagram for the BFIFO1 register



The following table shows the BFIFO1 register bit descriptions.

Table 9-453: BFIFO1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	Byte0	UNKNOWN	RW	First byte.

9.17.5 css600_jtagap Byte FIFO Register 2, BFIFO2

The BFIFO2 register enables 2 bytes to be transacted with the Response or Command FIFO.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD14

Type

RW

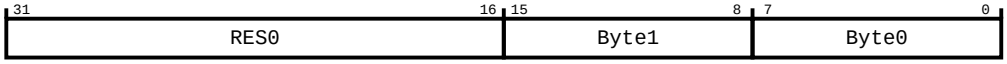
Reset value

0x0000----

Bit descriptions

The following figure shows the BFIFO2 register bit assignments.

Figure 9-438: Bit assignment diagram for the BFIFO2 register



The following table shows the BFIFO2 register bit descriptions.

Table 9-454: BFIFO2 bit descriptions

Bits	Name	Reset	Type	Description
31:16	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
15:8	Byte1	**UNKNOWN**	RW	Second byte.
7:0	Byte0	**UNKNOWN**	RW	First byte.

9.17.6 css600_jtagap Byte FIFO Register 3, BFIFO3

The BFIFO3 register enables 3 bytes to be transacted with the Response or Command FIFO.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xD18

Type

RW

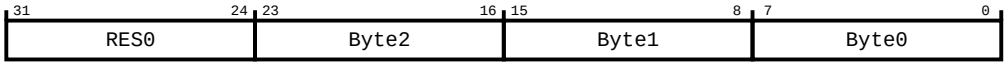
Reset value

0x00------

Bit descriptions

The following figure shows the BFIFO3 register bit assignments.

Figure 9-439: Bit assignment diagram for the BFIFO3 register



The following table shows the BFIFO3 register bit descriptions.

Table 9-455: BFIFO3 bit descriptions

Bits	Name	Reset	Type	Description
31:24	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
23:16	Byte2	**UNKNOWN**	RW	Third byte.
15:8	Byte1	**UNKNOWN**	RW	Second byte.
7:0	Byte0	**UNKNOWN**	RW	First byte.

9.17.7 css600_jtagap Byte FIFO Register 4, BFIFO4

The BFIFO4 register enables 4 bytes to be transacted with the Response or Command FIFO.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xD1C`

Type

RW

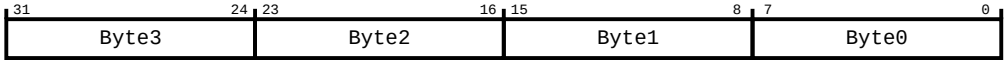
Reset value

`0x--------`

Bit descriptions

The following figure shows the BFIFO4 register bit assignments.

Figure 9-440: Bit assignment diagram for the BFIFO4 register



The following table shows the BFIFO4 register bit descriptions.

Table 9-456: BFIFO4 bit descriptions

Bits	Name	Reset	Type	Description
31:24	Byte3	UNKNOWN	RW	Forth byte.
23:16	Byte2	UNKNOWN	RW	Third byte.
15:8	Byte1	UNKNOWN	RW	Second byte.
7:0	Byte0	UNKNOWN	RW	First byte.

9.17.8 css600_jtagap Identification Register, IDR

This register provides a mechanism for the debugger to know various identity attributes of the AP.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xDFC

Type

RO

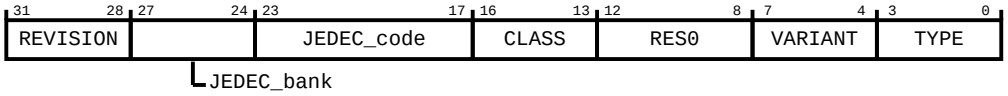
Reset value

0x44760020

Bit descriptions

The following figure shows the IDR register bit assignments.

Figure 9-441: Bit assignment diagram for the IDR register



The following table shows the IDR register bit descriptions.

Table 9-457: IDR bit descriptions

Bits	Name	Reset	Type	Description
31:28	REVISION	0b0100	RO	Revision. An incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
27:24	JEDEC_bank	0b0100	RO	The JEP106 continuation code. **0b0100** Arm
23:17	JEDEC_code	0x3B	RO	The JEP106 identification code. **0x3B** Arm
16:13	CLASS	0b0000	RO	Defines the class of the AP. **0b0000** No defined class
12:8	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
7:4	VARIANT	0b0010	RO	Together with the TYPE field, this field identifies the AP implementation. VARIANT differentiates AP implementations that have the same value of TYPE.
3:0	TYPE	0b0000	RO	Indicates the type of bus, or other connection, that connects to the AP. **0b0000** JTAG connection.

9.17.9 css600_jtagap Integration Test Status register, ITSTATUS

This register indicates the Integration Test DP Abort status.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEFC`

Type

RO

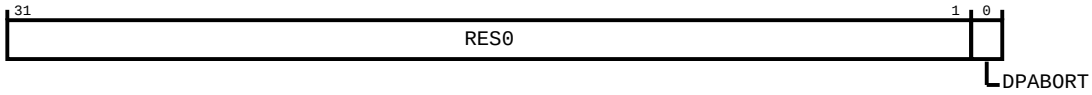
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITSTATUS register bit assignments.

Figure 9-442: Bit assignment diagram for the ITSTATUS register



The following table shows the ITSTATUS register bit descriptions.

Table 9-458: ITSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
0	DPABORT	`0b0`	RO	When in Integration testing mode (ITCTRL.IME=`0b1`): Behaves as a sticky bit and latches to 1 on a rising edge of dp_abort. Cleared on a read from this register. If dp_abort rises in the same cycle as a read of the ITSTATUS register is received, the read takes priority and the register is cleared. When in normal functional operation mode (ITCTRL.IME=`0b0`): Read as 0, writes ignored.

9.17.10 css600_jtagap Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xF00`

Type

RW

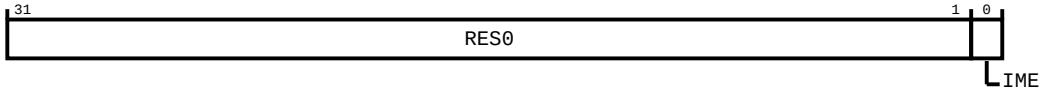
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-443: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-459: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
0	IME	`0b0`	RW	Integration Mode Enable. **`0b0`** The component must enter functional mode. **`0b1`** The component must enter integration mode, and enable support for topology detection and integration testing.

9.17.11 css600_jtagap Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

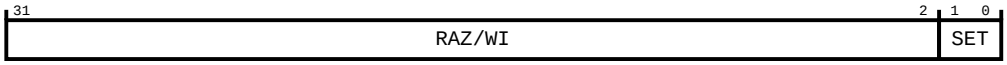
Reset value

0x00000003

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-444: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-460: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:2	RAZ/WI	0x0	RO	RAZ/WI.
1:0	SET	0b11	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.17.12 css600_jtagap Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA4

Type

RW

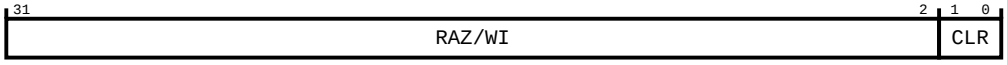
Reset value

0x00000000

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-445: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-461: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:2	RAZ/WI	0x0	RO	RAZ/WI.
1:0	CLR	0b00	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.17.13 css600_jtagap Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFBC

Type

RO

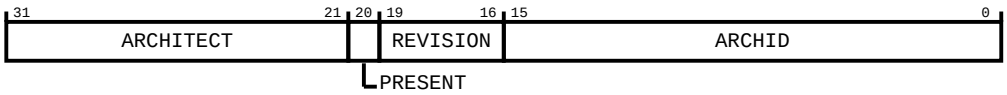
Reset value

0x47700A27

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-446: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-462: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	0x23B	RO	Defines the architect of the component **0x23B** Arm
20	PRESENT	0b1	RO	Indicates the presence of this register **0b1** DEVARCH is present
19:16	REVISION	0b0000	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	0x0A27	RO	Architecture ID. Returns a value that identifies the architecture of the component. **0x0A27** JTAG Access Port v2 architecture

9.17.14 css600_jtagap Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

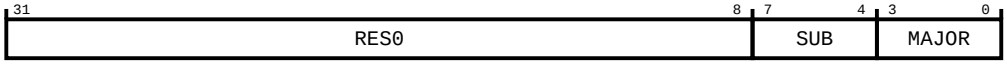
Reset value

0x00000000

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-447: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-463: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	0b0000	RO	Minor classification. Returns 0x0, Other/undefined.
3:0	MAJOR	0b0000	RO	Major classification. Returns 0x0, Miscellaneous.

9.17.15 css600_jtagap Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

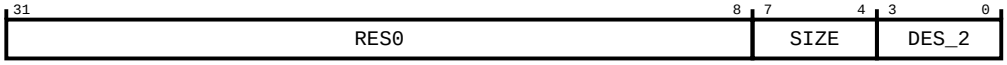
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-448: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-464: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.17.16 css600_jtagap Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD4`

Type

RO

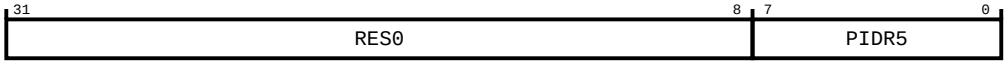
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-449: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-465: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.17.17 css600_jtagap Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

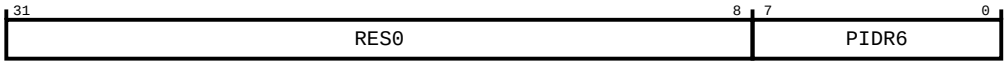
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-450: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-466: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.17.18 css600_jtagap Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

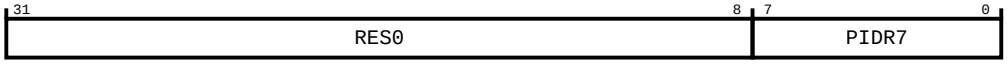
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-451: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-467: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.17.19 css600_jtagap Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

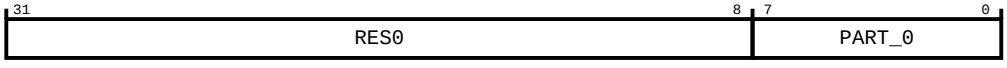
Reset value

0x000000E6

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-452: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-468: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:0	PART_0	0xE6	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.17.20 css600_jtagap Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

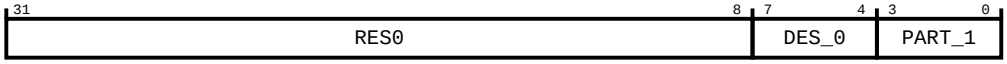
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-453: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-469: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.17.21 css600_jtagap Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFE8`

Type

RO

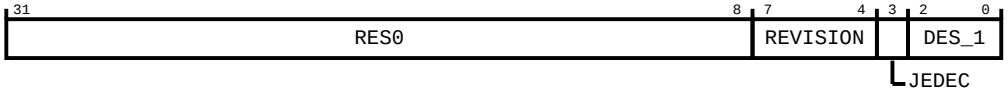
Reset value

`0x0000004B`

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-454: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-470: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	`0b0100`	RO	Revision. It is an incremental value starting at `0x0` for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	`0b1`	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	`0b011`	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.17.22 css600_jtagap Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

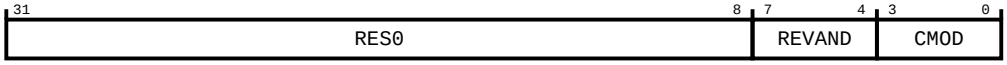
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-455: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-471: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	`0b0000`	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is `0x0`.
3:0	CMOD	`0b0000`	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is `0x0`.

9.17.23 css600_jtagap Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF0`

Type

RO

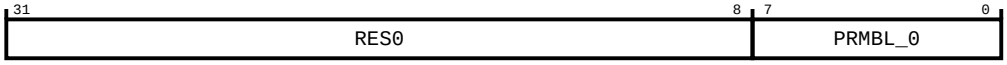
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-456: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-472: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.17.24 css600_jtagap Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

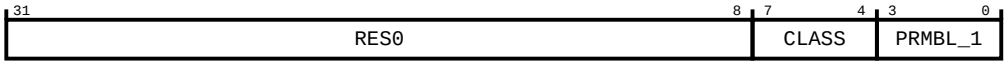
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-457: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-473: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.17.25 css600_jtagap Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

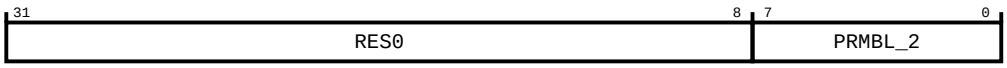
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-458: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-474: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.17.26 css600_jtagap Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFFC`

Type

RO

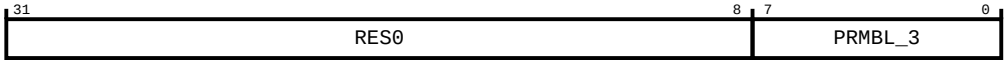
Reset value

`0x000000B1`

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-459: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-475: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.18 css600_tmc_etb register summary

This section describes the css600_tmc_etb_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-476: css600_tmc_etb_registers register summary

Offset	Name	Type	Reset	Width	Description
0x004	RSZ	RO	0x--------	32-bit	RAM Size register
0x00c	STS	RO	0x0000001-	32-bit	Status register
0x010	RRD	RO	0x--------	32-bit	RAM Read Data register

Offset	Name	Type	Reset	Width	Description
0x014	RRP	RW	0x--------	32-bit	RAM Read Pointer register
0x018	RWP	RW	0x--------	32-bit	RAM Write Pointer register
0x01c	TRG	RW	0x--------	32-bit	Trigger Counter register
0x020	CTL	RW	0x00000000	32-bit	Control Register
0x024	RWD	WO	0x00000000	32-bit	RAM Write Data register
0x028	MODE	RW	0x000000--	32-bit	Mode register
0x02c	LBUFLEVEL	RO	0x--------	32-bit	Latched Buffer Fill Level
0x030	CBUFLEVEL	RO	0x--------	32-bit	Current Buffer Fill Level
0x034	BUFWM	RW	0x--------	32-bit	Buffer Level Water Mark
0x300	FFSR	RO	0x0000000-	32-bit	Formatter and Flush Status Register
0x304	FFCR	RW	0x00000000	32-bit	Formatter and Flush Control Register
0x308	PSCR	RW	0x0000000A	32-bit	Periodic Synchronization Counter Register
0xee0	ITEVTINTR	WO	0x00000000	32-bit	Integration Test Event and Interrupt Control Register
0xee8	ITTRFLIN	RO	0x00000000	32-bit	Integration Test Trigger In and Flush In register
0xeec	ITATBDATA0	RO	0x00000000	32-bit	Integration Test ATB Data 0 Register
0xef0	ITATBCTR2	WO	0x00000000	32-bit	Integration Test ATB Control 2 Register
0xef4	ITATBCTR1	RO	0x00000000	32-bit	Integration Test ATB Control 1 Register
0xef8	ITATBCTR0	RO	0x00000000	32-bit	Integration Test ATB Control 0 Register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x0000000F	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x00000000	32-bit	Authentication Status Register
0xfc4	DEVID1	RO	0x00000001	32-bit	Device Configuration Register 1
0xfc8	DEVID	RO	0x00000-00	32-bit	Device Configuration Register
0xfcc	DEVTYPE	RO	0x00000021	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000E9	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000009B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.18.1 css600_tmc_etb RAM Size register, RSZ

The RSZ register defines the size of trace memory in units of 32-bit words.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x004`

Type

RO

Reset value

`0x--------`

Bit descriptions

The following figure shows the RSZ register bit assignments.

Figure 9-460: Bit assignment diagram for the RSZ register

The following table shows the RSZ register bit descriptions.

Table 9-477: RSZ bit descriptions

Bits	Name	Reset	Type	Description
31	**RES0**	`0b0`	RO	Reserved bit or field with SBZP behavior.
30:0	RSZ	**IMPLEMENTATION DEFINED**	RO	RAM size. Indicates the size of the RAM in 32-bit words. For example: Returns `0x00000100` if trace memory size is 1KB, `0x40000000` if trace memory size is 4GB. This field has the same value as the MEM_SIZE parameter.

9.18.2 css600_tmc_etb Status register, STS

The STS register indicates the status of the Trace Memory Controller. After a reset, software must ignore all the fields of this register except STS.TMCReady. The other fields only have meaning when the TMC has left the Disabled state. Writes to all RO fields of this register are ignored.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x00C

Type

RO

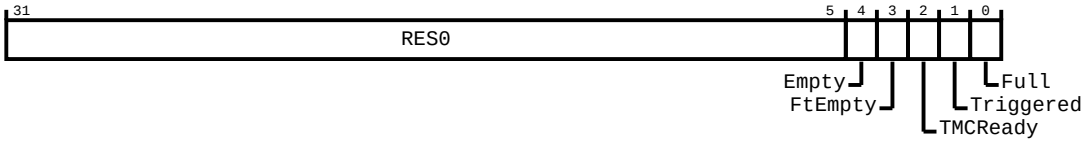
Reset value

0x0000001-

Bit descriptions

The following figure shows the STS register bit assignments.

Figure 9-461: Bit assignment diagram for the STS register



The following table shows the STS register bit descriptions.

Table 9-478: STS bit descriptions

Bits	Name	Reset	Type	Description
31:5	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
4	Empty	0b1	RO	Trace buffer empty. If set, this bit indicates that the trace memory does not contain any valid trace data. The reset value of this bit is 1. On leaving Disabled state, this bit dynamically indicates the empty status of trace memory, CBUFLEVEL == 0. However, this does not mean that the pipeline stages within the TMC are empty. To determine whether the internal pipeline stages are empty, the software must read the STS.TMCReady bit. In Circular Buffer mode the Empty bit and the Full bit in this register can be 1 at the same time. This happens because the Full bit in this mode, when set, does not clear until TraceCaptEn is set.
3	FtEmpty	0b1	RO	Trace capture has been completed and all captured trace data has been written to the trace memory, set in Stopped or Disabled state. Otherwise, it is cleared. The reset value is 1
2	TMCReady	0b1	RO	Trace capture has been completed and all captured trace data has been written to the trace memory
1	Triggered	**UNKNOWN**	RO	TMC triggered. This bit is set when trace capture is in progress and the TMC has detected a trigger event. This bit is cleared to 0 when leaving the Disabled state and retains its value when entering the Disabled state. A trigger event is when the TMC has written a set number of data words, as programmed in the TRG register, into the trace memory after a rising edge of trigin input, or a trigger packet (atid_rx = 0x7D) is received in the input trace.
0	Full	**UNKNOWN**	RO	Trace memory full. This bit helps determine the amount of valid data present in the trace memory. It is not affected by the reprogramming of pointer registers in Disabled state. It is cleared when the TMC leaves Disabled state. In Circular Buffer mode, this flag is set when the RAM write pointer wraps around the top of the buffer. It remains set until the TraceCaptEn bit is cleared and set. In Software FIFO mode, this flag indicates that the current space in the trace memory is less than or equal to the value programmed in the BUFWM Register, that is, Fill level >= MEM_SIZEBUFWM. The FULL output from the TMC reflects the value of this register bit, except when the Integration Mode bit in the ITCTRL Register, 0xF00, is set.

9.18.3 css600_tmc_etb RAM Read Data register, RRD

Reading this register allows data to be read from the trace memory at the location pointed to by the RRP register when either in the Disabled state or operating in Circular Buffer mode (CB) or Software FIFO mode 1 (SWF1).

When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the memory width is twice as wide and a memory word holds 8, 16 or 32 bytes. Multiple RRD reads must be performed to read a full memory word. When a full memory width of data has been read via the RRD register, the RRP register is incremented to the next memory word. When the TMC left the Disabled state and the trace memory is empty, this register returns 0xFFFFFFFF. When the TMC left the Disabled state and the trace memory is empty, this register returns 0xFFFFFFFF. When operating in CB mode and the TMC left the Disabled state, this register returns 0xFFFFFFFF in all other states except the Stopped state.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x010`

Type

RO

Reset value

`0x--------`

Bit descriptions

The following figure shows the RRD register bit assignments.

Figure 9-462: Bit assignment diagram for the RRD register

The following table shows the RRD register bit descriptions.

Table 9-479: RRD bit descriptions

Bits	Name	Reset	Type	Description
31:0	RRD	UNKNOWN	RO	Returns the data read from trace memory

9.18.4 css600_tmc_etb RAM Read Pointer register, RRP

The RRP register contains the value of the read pointer that is used to read entries from trace memory over the APB interface. Software must program it with the same value as RWP before enabling trace capture.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x014`

Type

RW

Reset value

`0x--------`

Bit descriptions

The following figure shows the RRP register bit assignments.

Figure 9-463: Bit assignment diagram for the RRP register

The following table shows the RRP register bit descriptions.

Table 9-480: RRP bit descriptions

Bits	Name	Reset	Type	Description
31:0	RRP	**UNKNOWN**	RW	The RRP width depends on the size of trace memory and is given by log2(MEM_SIZE x 4). The remaining MSBs of the 32-bit register are of type **RAZ/WI**. When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the memory width is twice as wide and a memory word holds 8, 16 or 32 bytes. When a full memory width of data has been read via the RRD register, the RRP register is incremented to the next memory word. The lowest 4 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 32 or 64 bits. The lowest 5 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 128 bits.

9.18.5 css600_tmc_etb RAM Write Pointer register, RWP

The RWP register sets the write pointer that writes entries into the trace memory. Software must program it before enabling trace capture.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x018

Type

RW

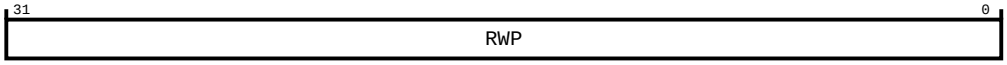
Reset value

0x--------

Bit descriptions

The following figure shows the RWP register bit assignments.

Figure 9-464: Bit assignment diagram for the RWP register



The following table shows the RWP register bit descriptions.

Table 9-481: RWP bit descriptions

Bits	Name	Reset	Type	Description
31:0	RWP	**UNKNOWN**	RW	The RWP width depends on the size of trace memory and is given by log2(MEM_SIZE x 4). The remaining MSBs of the 32-bit register are of type **RAZ/WI**. When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the memory width is twice as wide and a memory word holds 8, 16 or 32 bytes. When a full memory width of data has been written to the RWD register, the RWP register is incremented to the next memory word. The lowest 4 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 32 or 64 bits. The lowest 5 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 128 bits.

9.18.6 css600_tmc_etb Trigger Counter register, TRG

The TRG register, in Circular Buffer mode, specifies the number of 32-bit words to capture in the trace memory, after detecting either a rising edge on the trigin input or a trigger packet in the incoming trace stream, that is, where atid_rx = 0x7D.

The value programmed must be aligned to the frame length of 128 bits. Software must program this register before leaving Disabled state.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x01C

Type

RW

Reset value

0x--------

Bit descriptions

The following figure shows the TRG register bit assignments.

Figure 9-465: Bit assignment diagram for the TRG register

The following table shows the TRG register bit descriptions.

Table 9-482: TRG bit descriptions

Bits	Name	Reset	Type	Description
31:0	TRG	UNKNOWN	RW	Trigger count. This count represents the number of 32-bit words of trace that are captured between a trigger packet and a trigger event. The lowest two bits have access type **RAZ/WI**.

9.18.7 css600_tmc_etb Control Register, CTL

The CTL register controls trace stream capture. Setting the CTL.TraceCaptEn bit to 1 enables the TMC to capture the trace data. When trace capture is enabled, the formatter behavior is controlled by the FFCR register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x020

Type

RW

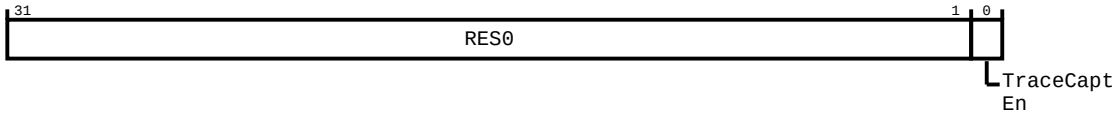
Reset value

0x00000000

Bit descriptions

The following figure shows the CTL register bit assignments.

Figure 9-466: Bit assignment diagram for the CTL register



The following table shows the CTL register bit descriptions.

Table 9-483: CTL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	TraceCaptEn	0b0	RW	Trace capture enable: **0b0** Disable trace capture **0b1** Enable trace capture

9.18.8 css600_tmc_etb RAM Write Data register, RWD

The RWD register enables testing of trace memory connectivity to the TMC. Writing this register allows data to be written to the trace memory at the location pointed to by the RWP register when in the Disabled state.

When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the memory width is twice as wide and a memory word holds 8, 16 or 32 bytes. Multiple RWD writes must be performed to write a full memory word. When a full memory width of data has been written via the RWD register, the data is written to the trace memory and the RWP register is incremented to the next memory word.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x024

Type

WO

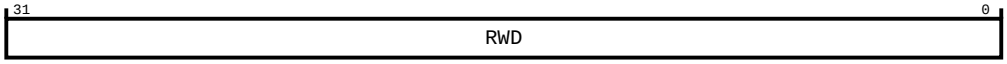
Reset value

0x00000000

Bit descriptions

The following figure shows the RWD register bit assignments.

Figure 9-467: Bit assignment diagram for the RWD register



The following table shows the RWD register bit descriptions.

Table 9-484: RWD bit descriptions

Bits	Name	Reset	Type	Description
31:0	RWD	0x0	WO	Data written to this register is placed in the trace memory.

9.18.9 css600_tmc_etb Mode register, MODE

The MODE register controls the TMC operating mode. The operating mode can only be changed when the TMC is in Disabled state. Attempting to write to this register in any other state results in **UNPREDICTABLE** behavior. The operating mode is ignored when in Disabled state.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x028

Type

RW

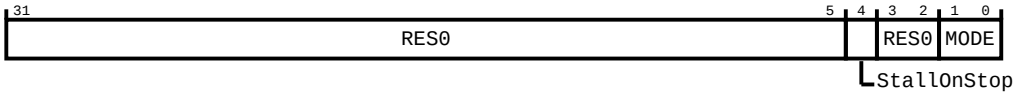
Reset value

0x000000--

Bit descriptions

The following figure shows the MODE register bit assignments.

Figure 9-468: Bit assignment diagram for the MODE register



The following table shows the MODE register bit descriptions.

Table 9-485: MODE bit descriptions

Bits	Name	Reset	Type	Description
31:5	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
4	StallOnStop	UNKNOWN	RW	Stall On Stop. If this bit is set and the formatter stops as a result of a stop event, the output atready_rx is de-asserted to stall the ATB interface and avoid loss of trace. If this bit is clear and the formatter stops as a result of a stop event, signal atready_rx remains asserted but the TMC discards further incoming trace.
3:2	RES0	0b00	RO	Reserved bit or field with SBZP behavior.
1:0	MODE	UNKNOWN	RW	Selects the operating mode after leaving Disabled state. If a reserved MODE value is programmed and trace capture is enabled, the TMC starts to operate in Software FIFO mode 1 (SWF1). However, reading the MODE.MODE field returns the programmed value. **0b00** CB, Circular Buffer mode. **0b01** SWF1, Software Read FIFO mode 1. **0b10** Reserved. (SWF1) **0b11** Reserved. (SWF1)

9.18.10 css600_tmc_etb Latched Buffer Fill Level, LBUFLEVEL

Reading the LBUFLEVEL register returns the maximum fill level of the trace memory in 32-bit words since this register was last read. Reading this register also results in its contents being updated to the current fill level. When entering Disabled state, it retains its last value.

While in Disabled state, reads from this register do not affect its value. When exiting Disabled state, the LBUFLEVEL register is updated to the current fill level.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x02C

Type

RO

Reset value

0x--------

Bit descriptions

The following figure shows the LBUFLEVEL register bit assignments.

Figure 9-469: Bit assignment diagram for the LBUFLEVEL register

The following table shows the LBUFLEVEL register bit descriptions.

Table 9-486: LBUFLEVEL bit descriptions

Bits	Name	Reset	Type	Description
31:0	LBUFLEVEL	**UNKNOWN**	RO	Latched Buffer Fill Level. Indicates the maximum fill level of the trace memory in 32-bit words since this register was last read. The width of the register is 1 + log2(MEM_SIZE).

9.18.11 css600_tmc_etb Current Buffer Fill Level, CBUFLEVEL

The CBUFLEVEL register indicates the current fill level of the trace memory in units of 32-bit words. When the TMC leaves Disabled state, this register dynamically indicates the current fill level of trace memory. It retains its value on entering Disabled state.

It is not affected by the reprogramming of pointer registers in Disabled state with the exception of RRD reads and RWD writes. Before leaving the Disabled state software must program RRP with the same value as RWP. Without doing this results in **UNPREDICTABLE** behavior.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x030`

Type

RO

Reset value

`0x--------`

Bit descriptions

The following figure shows the CBUFLEVEL register bit assignments.

Figure 9-470: Bit assignment diagram for the CBUFLEVEL register

The following table shows the CBUFLEVEL register bit descriptions.

Table 9-487: CBUFLEVEL bit descriptions

Bits	Name	Reset	Type	Description
31:0	CBUFLEVEL	UNKNOWN	RO	Current Buffer Fill Level. Indicates the current fill level of the trace memory in 32-bit words. The width of the register is 1 + log2(MEM_SIZE).

9.18.12 css600_tmc_etb Buffer Level Water Mark, BUFWM

The value that is programmed into the BUFWM register indicates the required threshold vacancy level in 32-bit words in the trace memory.

When the available space in the FIFO is less than or equal to this value, that is, fill level >= (MEM_SIZE - BUFWM), the full output is asserted and the STS.Full bit is set. This register is used only in the FIFO modes, that is, Software FIFO mode 1 (SWF1), Software FIFO mode 2 (SWF2), and Hardware FIFO (HWF) modes. In Circular Buffer (CB) mode, the same functionality is obtained by programming the RWP to the required vacancy trigger level, so that when the pointer wraps around, the full output gets asserted indicating that the vacancy level has fallen below the required level. Reading this register returns the programmed value. The maximum value that can be written into this register is MEM_SIZE - 1, in which case the full output is asserted after the first 32-bit word is written to trace memory. Writing to this register other than when in Disabled state results in **UNPREDICTABLE** behavior. Any software using it must program it with an initial value before setting the CTL.TraceCaptEn bit to 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x034`

Type

RW

Reset value

`0x--------`

Bit descriptions

The following figure shows the BUFWM register bit assignments.

Figure 9-471: Bit assignment diagram for the BUFWM register

The following table shows the BUFWM register bit descriptions.

Table 9-488: BUFWM bit descriptions

Bits	Name	Reset	Type	Description
31:0	BUFWM	UNKNOWN	RW	Buffer Level Watermark. Indicates the desired threshold vacancy level in 32-bit words in the trace memory. The width of the register is log2(MEM_SIZE).

9.18.13 css600_tmc_etb Formatter and Flush Status Register, FFSR

This register indicates the status of the Formatter, and the status of Flush request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x300`

Type

RO

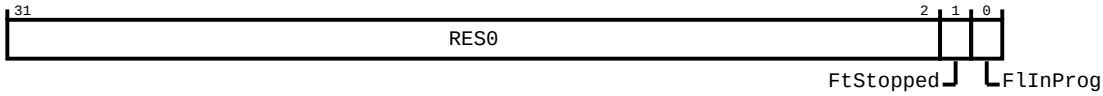
Reset value

`0x0000000-`

Bit descriptions

The following figure shows the FFSR register bit assignments.

Figure 9-472: Bit assignment diagram for the FFSR register



The following table shows the FFSR register bit descriptions.

Table 9-489: FFSR bit descriptions

Bits	Name	Reset	Type	Description
31:2	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
1	FtStopped	UNKNOWN	RO	Formatter Stopped. This bit behaves the same way as STS.FtEmpty.
0	FlInProg	UNKNOWN	RO	Flush In Progress. This bit indicates whether the TMC is currently processing a flush request. The flush initiation is controlled by the flush control bits in the FFCR register. This bit is cleared to 0 when leaving the Disabled state and retains its value when entering the Disabled state. When in Disabled state, this bit is not updated. **0b0** No flush activity in progress. **0b1** Flush in progress on the ATB receiver interface or the TMC internal pipeline.

9.18.14 css600_tmc_etb Formatter and Flush Control Register, FFCR

The FFCR controls the generation of stop, trigger and flush events. The insertion of a flush completion packet and the insertion of a trigger packet in the formatted trace is enabled here.

Also one of the 2 formatter modes for bypass mode and normal mode can be changed here when the formatter has stopped.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x304`

Type

RW

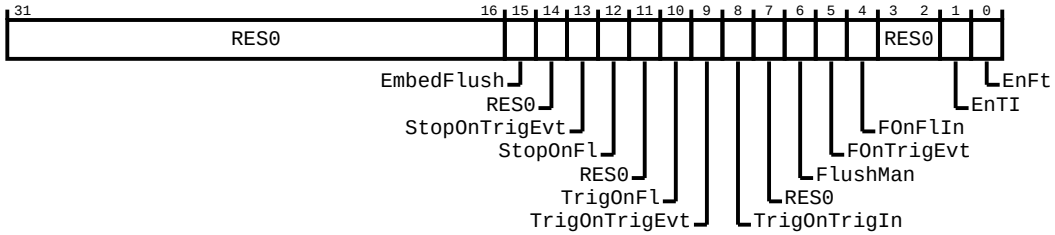
Reset value

`0x00000000`

Bit descriptions

The following figure shows the FFCR register bit assignments.

Figure 9-473: Bit assignment diagram for the FFCR register



The following table shows the FFCR register bit descriptions.

Table 9-490: FFCR bit descriptions

Bits	Name	Reset	Type	Description
31:16	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15	EmbedFlush	0b0	RW	Embed Flush ID (flush completion packet). Enables insertion of Flush ID 0x7B with a single byte of data payload = 0x00 in the output trace, immediately after the last flush data byte, when a flush completes on the ATB receiver interface. This bit is effective only in Normal formatting modes. In Bypass mode, the Flush ID insertion remains disabled and this bit is ignored. **0b0** Disable Flush ID insertion. **0b1** Enable Flush ID insertion.
14	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
13	StopOnTrigEvt	0b0	RW	Stop On Trigger Event. If this bit is set, the formatter is stopped when a Trigger Event has been observed. This bit must be used only in CB mode. If trace capture is enabled in Software FIFO mode 1 (SWF1) with this bit set, it results in **UNPREDICTABLE** behavior.
12	StopOnFl	0b0	RW	Stop On Flush. If this bit is set, the formatter is stopped on completion of a flush operation. The initiation of a flush operation is controlled by programming the register bits FFCR.FlushMan, FFCR.FOnTrigEvt, and FFCR.FOnFlIn. When a flush-initiation condition occurs, afvalid_rx is asserted, and when the flush completion is received, that is, afready_rx=1, trace capture is stopped. Any remaining data in the formatter is appended with a post-amble and written to trace memory. The flush operation is then complete.
11	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
10	TrigOnFl	0b0	RW	Indicate on trace stream the completion of flush. If this bit is set, a trigger is indicated on the trace stream when afready_rx is received for a flush in progress. If this bit is clear, no triggers are embedded in the trace stream on flush completion. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit.
9	TrigOnTrigEvt	0b0	RW	Indicate on trace stream the occurrence of a Trigger Event. If this bit is set, a trigger is indicated on the output trace stream when a Trigger Event occurs. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit. This bit must be used only in CB mode. If trace capture is enabled in SWF1 mode with this bit set, it results in **UNPREDICTABLE** behavior.
8	TrigOnTrigIn	0b0	RW	Indicate on trace stream the occurrence of a rising edge on trigin. If this bit is set, a trigger is indicated on the trace stream when a rising edge is detected on the trigin input. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit.
7	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
6	FlushMan	0b0	RW	Manually generate a flush of the system. Writing 1 to this bit causes a flush to be generated. This bit is cleared automatically when, in formatter bypass mode, afready_rx was sampled high, or, in normal formatting mode, afready_rx was sampled high and all flush data was output to the trace memory. If CTL.TraceCaptEn=0, writes to this bit are ignored.
5	FOnTrigEvt	0b0	RW	Flush on Trigger Event. If FFCR.StopOnTrigEvt is set, this bit is ignored. Setting this bit generates a flush when a Trigger Event occurs. If FFCR.StopOnTrigEvt is set, this bit is ignored. This bit must be used only in CB mode. If trace capture is enabled in SWF1 mode with this bit set, it results in **UNPREDICTABLE** behavior.
4	FOnFlIn	0b0	RW	Setting this bit enables the detection of transitions on the flushin input by the TMC. If this bit is set and the formatter has not already stopped, a rising edge on flushin initiates a flush request.
3:2	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
1	EnTI	0b0	RW	Enable Trigger Insertion. Setting this bit enables the insertion of triggers in the formatted trace stream. A trigger is indicated by inserting one byte of data 0x00 with atid_rx=0x7D in the trace stream. Trigger indication on the trace stream is also controlled by the register bits FFCR.TrigOnFl, FFCR.TrigOnTrigEvt, and FFCR.TrigOnTrigIn. This bit can only be changed when the TMC is in Disabled state. If FFCR.EnTI bit is set formatting is enabled.

Bits	Name	Reset	Type	Description
0	EnFt	0b0	RW	Enable Formatter. If this bit is set, formatting is enabled. When EnTi is set, formatting is enabled. When CB mode is not used, formatting is also enabled. For backwards-compatibility with earlier versions of the ETB disabling of formatting is supported only in CB mode. This bit can only be changed when TMC is in Disabled state.

9.18.15 css600_tmc_etb Periodic Synchronization Counter Register, PSCR

This register determines the reload value of the Periodic Synchronization Counter. This counter enables the frequency of sync packets to be optimized to the trace capture buffer size. The default behavior of the counter is to generate periodic synchronization requests, syncreq_rx, on the ATB receiver interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x308

Type

RW

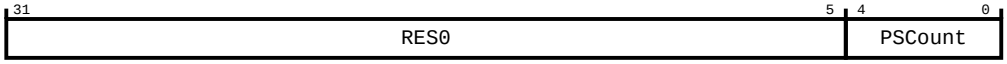
Reset value

0x0000000A

Bit descriptions

The following figure shows the PSCR register bit assignments.

Figure 9-474: Bit assignment diagram for the PSCR register



The following table shows the PSCR register bit descriptions.

Table 9-491: PSCR bit descriptions

Bits	Name	Reset	Type	Description
31:5	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
4:0	PSCount	0b01010	RW	Periodic Synchronization Count. Determines the reload value of the Synchronization Counter. The reload value takes effect the next time the counter reaches zero. When trace capture is enabled, the Synchronization Counter counts the number of bytes of trace data that is stored into the trace memory, regardless of whether the trace data has been formatted by the TMC or not, since the occurrence of the last sync request on the ATB receiver interface. When the counter reaches 0, a sync request is sent on the ATB receiver interface. Reads from this register return the reload value that is programmed in this register. This field resets to 0x0A, that is, the default sync period is 2^10 bytes. If a reserved value is programmed in this register field, the value 0x1B is used instead, and subsequent reads from this register also return 0x1B. The following constraints apply to the values written to the PSCount field: 0x0 - synchronization is disabled, 0x1-0x6 - reserved, 0x7-0x1B - synchronization period is 2^PSCount bytes. The smallest value 0x7 gives a sync period of 128 bytes. The maximum allowed value 0x1B gives a sync period of 2^27 bytes, 0x1C-0x1F - reserved.

9.18.16 css600_tmc_etb Integration Test Event and Interrupt Control Register, ITEVTINTR

This register controls the values of event and interrupt outputs in integration mode. In functional mode, this register behaves as **RAZ/WI**.

In integration mode, the value that is written to any bit of this register is driven on the output pin that is controlled by that bit and the reads return 0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEE0

Type

WO

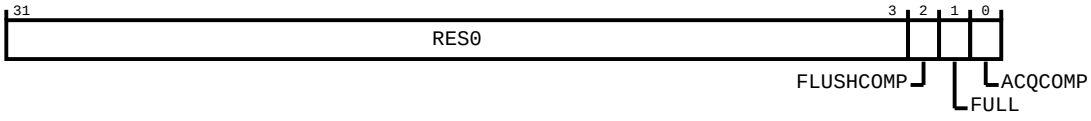
Reset value

0x00000000

Bit descriptions

The following figure shows the ITEVTINTR register bit assignments.

Figure 9-475: Bit assignment diagram for the ITEVTINTR register



The following table shows the ITEVTINTR register bit descriptions.

Table 9-492: ITEVTINTR bit descriptions

Bits	Name	Reset	Type	Description
31:3	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
2	FLUSHCOMP	0b0	WO	Controls the value of flushcomp output in integration mode.
1	FULL	0b0	WO	Controls the value of full output in integration mode.
0	ACQCOMP	0b0	WO	Controls the value of acqcomp output in integration mode.

9.18.17 css600_tmc_etb Integration Test Trigger In and Flush In register, ITTRFLIN

This register captures the values of the flushin and trigin inputs in integration mode. In functional mode, this register behaves as **RAZ/WI**.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEE8

Type

RO

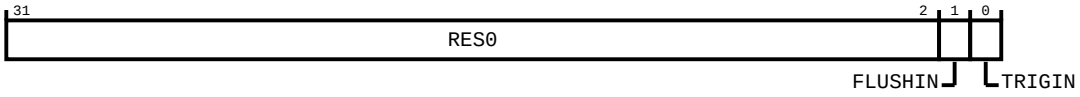
Reset value

0x00000000

Bit descriptions

The following figure shows the ITTRFLIN register bit assignments.

Figure 9-476: Bit assignment diagram for the ITTRFLIN register



The following table shows the ITTRFLIN register bit descriptions.

Table 9-493: ITTRFLIN bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	FLUSHIN	0b0	RO	Integration status of flushin input. In integration mode, this bit latches to 1 on a rising edge of the flushin input. It is cleared when the register is read or when integration mode is disabled.
0	TRIGIN	0b0	RO	Integration status of trigin input. In integration mode, this bit latches to 1 on a rising edge of the trigin input. It is cleared when the register is read or when integration mode is disabled.

9.18.18 css600_tmc_etb Integration Test ATB Data 0 Register, ITATBDATA0

This register captures the value of atdata_rx input in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of corresponding atdata_rx bits. The width of this register is given by: 1+(ATB DATA WIDTH)/8.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEEC`

Type

RO

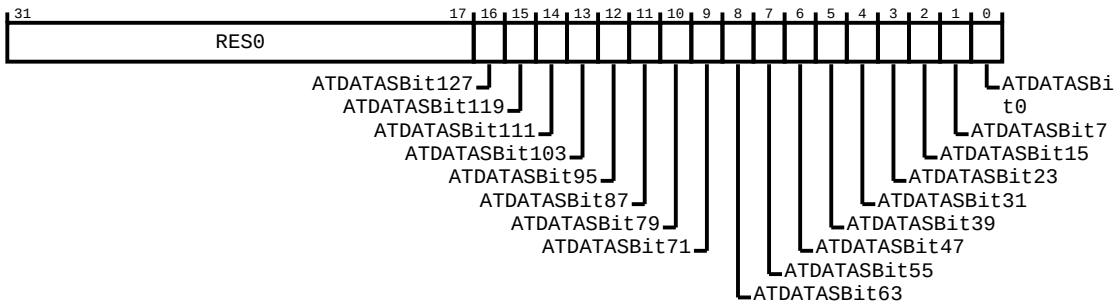
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITATBDATA0 register bit assignments.

Figure 9-477: Bit assignment diagram for the ITATBDATA0 register



The following table shows the ITATBDATA0 register bit descriptions.

Table 9-494: ITATBDATA0 bit descriptions

Bits	Name	Reset	Type	Description
31:17	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
16	ATDATASBit127	`0b0`	RO	Returns the value of atdata_rx[127] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
15	ATDATASBit119	`0b0`	RO	Returns the value of atdata_rx[119] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
14	ATDATASBit111	`0b0`	RO	Returns the value of atdata_rx[111] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.

Bits	Name	Reset	Type	Description
13	ATDATASBit103	0b0	RO	Returns the value of atdata_rx[103] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
12	ATDATASBit95	0b0	RO	Returns the value of atdata_rx[95] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
11	ATDATASBit87	0b0	RO	Returns the value of atdata_rx[87] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
10	ATDATASBit79	0b0	RO	Returns the value of atdata_rx[79] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
9	ATDATASBit71	0b0	RO	Returns the value of atdata_rx[71] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
8	ATDATASBit63	0b0	RO	Returns the value of atdata_rx[63] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
7	ATDATASBit55	0b0	RO	Returns the value of atdata_rx[55] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
6	ATDATASBit47	0b0	RO	Returns the value of atdata_rx[47] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
5	ATDATASBit39	0b0	RO	Returns the value of atdata_rx[39] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
4	ATDATASBit31	0b0	RO	Returns the value of atdata_rx[31] input in integration mode.
3	ATDATASBit23	0b0	RO	Returns the value of atdata_rx[23] input in integration mode.
2	ATDATASBit15	0b0	RO	Returns the value of atdata_rx[15] input in integration mode.
1	ATDATASBit7	0b0	RO	Returns the value of atdata_rx[7] input in integration mode.
0	ATDATASBit0	0b0	RO	Returns the value of atdata_rx[0] input in integration mode.

9.18.19 css600_tmc_etb Integration Test ATB Control 2 Register, ITATBCTR2

This register enables control of ATB receiver outputs atready_rx, afvalid_rx, and syncreq_rx in integration mode. In functional mode, this register behaves as **RAZ/WI**.

In integration mode, the value that is written to any bit of this register is driven on the output pin that is controlled by that bit and the reads return 0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF0

Type

WO

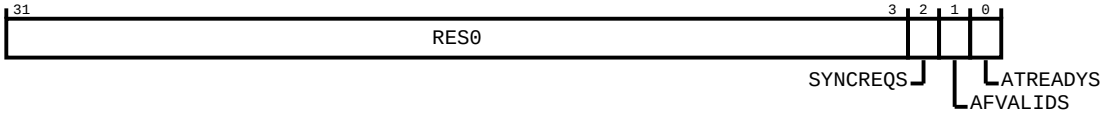
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR2 register bit assignments.

Figure 9-478: Bit assignment diagram for the ITATBCTR2 register



The following table shows the ITATBCTR2 register bit descriptions.

Table 9-495: ITATBCTR2 bit descriptions

Bits	Name	Reset	Type	Description
31:3	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
2	SYNCREQS	0b0	WO	Controls the value of syncreq_rx output in integration mode.
1	AFVALIDS	0b0	WO	Controls the value of afvalid_rx output in integration mode.
0	ATREADYS	0b0	WO	Controls the value of atready_rx output in integration mode.

9.18.20 css600_tmc_etb Integration Test ATB Control 1 Register, ITATBCTR1

This register captures the value of the atid_rx[6:0] input in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of atid_rx input.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF4

Type

RO

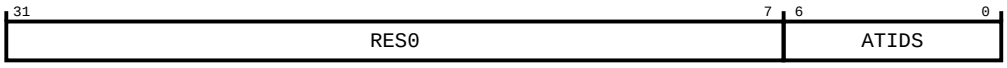
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR1 register bit assignments.

Figure 9-479: Bit assignment diagram for the ITATBCTR1 register



The following table shows the ITATBCTR1 register bit descriptions.

Table 9-496: ITATBCTR1 bit descriptions

Bits	Name	Reset	Type	Description
31:7	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
6:0	ATIDS	0x0	RO	Returns the value of atid_rx[6:0] input in integration mode.

9.18.21 css600_tmc_etb Integration Test ATB Control 0 Register, ITATBCTR0

This register captures the values of ATB receiver inputs atvalid_rx, afready_rx, atwakeup_rx, and atbytes_rx in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of corresponding input pins.

The width of this register is given by: 8+log2(ATB DATA WIDTH/8).

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF8

Type

RO

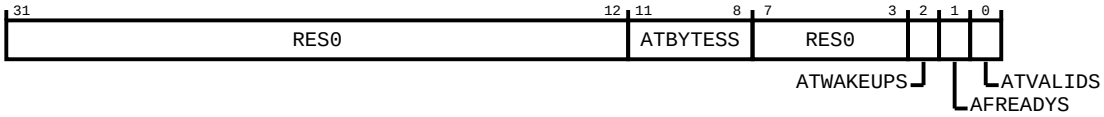
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR0 register bit assignments.

Figure 9-480: Bit assignment diagram for the ITATBCTR0 register



The following table shows the ITATBCTR0 register bit descriptions.

Table 9-497: ITATBCTR0 bit descriptions

Bits	Name	Reset	Type	Description
31:12	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
11:8	ATBYTESS	0b0000	RO	Returns the value of atbytes_rx input in integration mode. N=8+log2(ATB DATA WIDTH/8).

Bits	Name	Reset	Type	Description
7:3	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
2	ATWAKEUPS	0b0	RO	Returns the value of atwakeup_rx input in integration mode.
1	AFREADYS	0b0	RO	Returns the value of afready_rx input in integration mode.
0	ATVALIDS	0b0	RO	Returns the value of atvalid_rx input in integration mode.

9.18.22 css600_tmc_etb Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

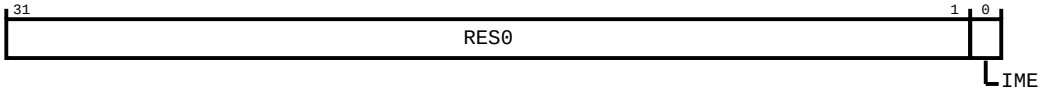
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-481: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-498: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.18.23 css600_tmc_etb Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

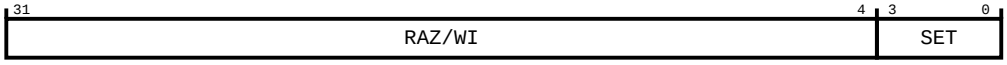
Reset value

0x0000000F

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-482: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-499: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:4	RAZ/WI	0x0	RO	RAZ/WI.
3:0	SET	0b1111	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.18.24 css600_tmc_etb Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFA4`

Type

RW

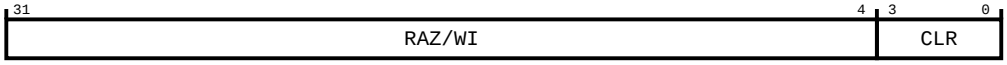
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-483: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-500: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	`0x0`	RO	**RAZ/WI**.
3:0	CLR	`0b0000`	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.18.25 css600_tmc_etb Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

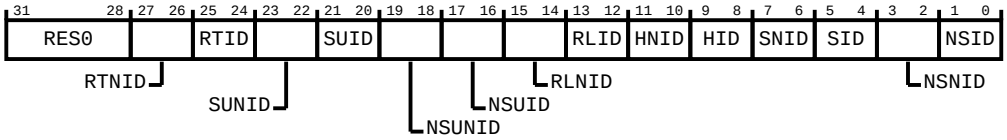
Reset value

0x00000000

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-484: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-501: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	0b00	RO	Root non-invasive debug. **0b00** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.
25:24	RTID	0b00	RO	Root invasive debug. **0b00** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	0b00	RO	Secure non-invasive debug. **0b00** Debug level is not supported.
5:4	SID	0b00	RO	Secure invasive debug. **0b00** Debug level is not supported.
3:2	NSNID	0b00	RO	Non-secure non-invasive debug. **0b00** Debug level is not supported.
1:0	NSID	0b00	RO	Non-secure invasive debug. **0b00** Debug level is not supported.

9.18.26 css600_tmc_etb Device Configuration Register 1, DEVID1

Contains an **IMPLEMENTATION DEFINED** value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC4

Type

RO

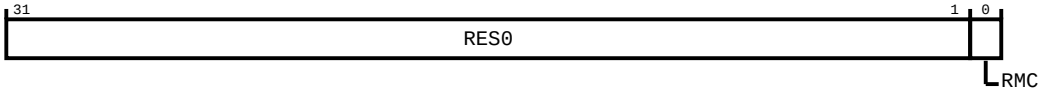
Reset value

0x00000001

Bit descriptions

The following figure shows the DEVID1 register bit assignments.

Figure 9-485: Bit assignment diagram for the DEVID1 register



The following table shows the DEVID1 register bit descriptions.

Table 9-502: DEVID1 bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	RMC	0b1	RO	Register management mode. TMC implements register management mode 1.

9.18.27 css600_tmc_etb Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC8

Type

RO

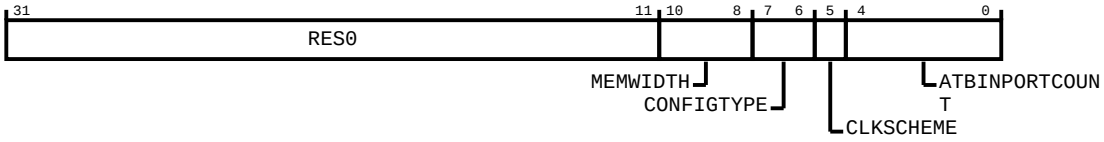
Reset value

0x00000-00

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-486: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-503: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:11	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
10:8	MEMWIDTH	**IMPLEMENTATION DEFINED**	RO	Indicates the width of the internal memory data bus. For the ETB this value is twice ATB_DATA_WIDTH. **0b011** Memory interface databus is 64 bits wide. (ATB_DATA_WIDTH = 32bit) **0b100** Memory interface databus is 128 bits wide. (ATB_DATA_WIDTH = 64bit) **0b101** Memory interface databus is 256 bits wide. (ATB_DATA_WIDTH = 128bit)
7:6	CONFIGTYPE	0b00	RO	Indicates the TMC configuration. **0b00** ETB - Embedded Trace Buffer
5	CLKSCHEME	0b0	RO	RAM Clocking Scheme. This value indicates the TMC RAM clocking scheme used, that is, whether the TMC RAM operates synchronously or asynchronously to the TMC clock. Fixed to 0 indicating that TMC RAM clock is synchronous to the clk input.
4:0	ATBINPORTCOUNT	0b00000	RO	Hidden Level of ATB input multiplexing. This value indicates the type/number of ATB multiplexing present on the input ATB. Fixed to 0x00 indicating that no multiplexing is present.

9.18.28 css600_tmc_etb Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

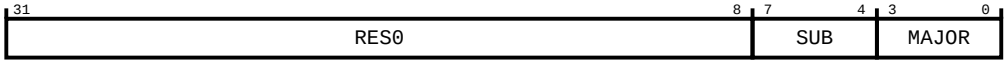
Reset value

0x00000021

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-487: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-504: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	0b0010	RO	Minor classification. Returns 0x2, indicating this component is a Buffer.
3:0	MAJOR	0b0001	RO	Major classification. Returns 0x1, indicating this component is a Trace Sink.

9.18.29 css600_tmc_etb Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

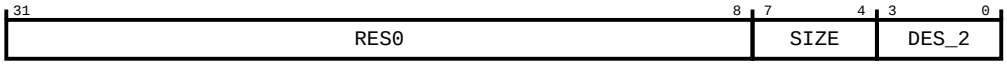
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-488: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-505: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.18.30 css600_tmc_etb Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

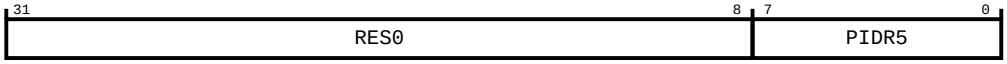
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-489: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-506: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.18.31 css600_tmc_etb Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

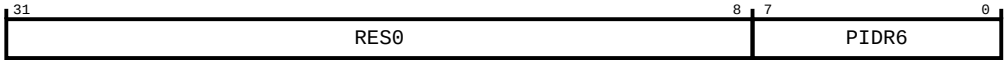
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-490: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-507: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.18.32 css600_tmc_etb Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFDC`

Type

RO

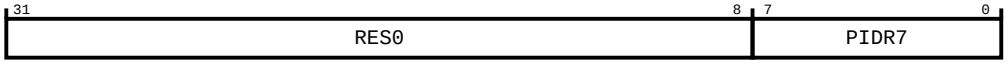
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-491: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-508: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.18.33 css600_tmc_etb Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

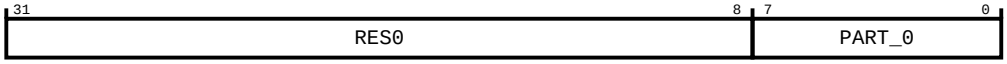
Reset value

0x000000E9

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-492: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-509: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xE9	RO	Part number (lower 8 bits). **0xE9** ETB - Embedded Trace Buffer

9.18.34 css600_tmc_etb Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

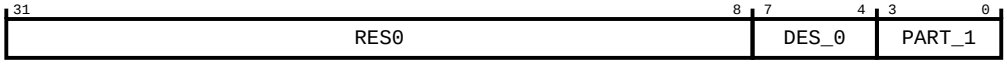
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-493: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-510: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.18.35 css600_tmc_etb Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

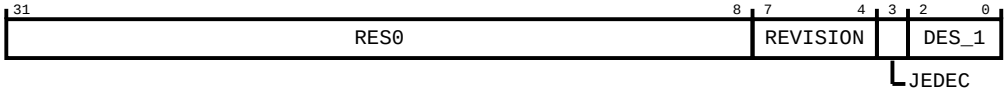
Reset value

0x0000009B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-494: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-511: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b1001	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.

Bits	Name	Reset	Type	Description
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.18.36 css600_tmc_etb Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFEC

Type

RO

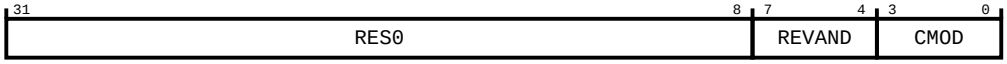
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-495: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-512: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.18.37 css600_tmc_etb Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF0`

Type

RO

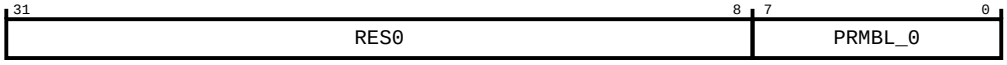
Reset value

`0x0000000D`

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-496: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-513: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	`0xD`	RO	Preamble. Returns `0x0D`.

9.18.38 css600_tmc_etb Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF4`

Type

RO

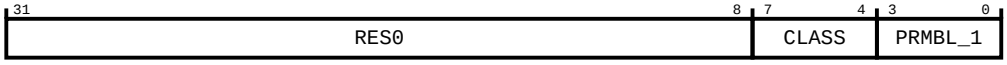
Reset value

`0x00000090`

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-497: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-514: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	`0b1001`	RO	Component class **`0b1001`** CoreSight component
3:0	PRMBL_1	`0b0000`	RO	Preamble. Returns `0x0`.

9.18.39 css600_tmc_etb Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF8`

Type

RO

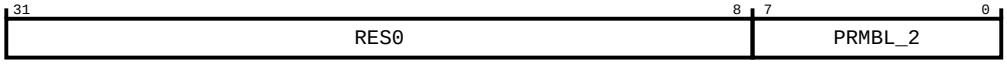
Reset value

`0x00000005`

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-498: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-515: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.18.40 css600_tmc_etb Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

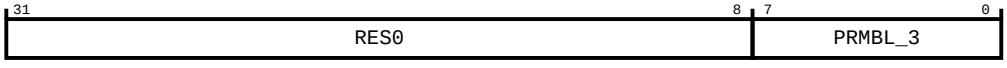
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-499: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-516: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.19 css600_tmc_etf register summary

This section describes the css600_tmc_etf_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-517: css600_tmc_etf_registers register summary

Offset	Name	Type	Reset	Width	Description
0x004	RSZ	RO	0x--------	32-bit	RAM Size register
0x00c	STS	RO	0x0000001-	32-bit	Status register
0x010	RRD	RO	0x--------	32-bit	RAM Read Data register
0x014	RRP	RW	0x--------	32-bit	RAM Read Pointer register
0x018	RWP	RW	0x--------	32-bit	RAM Write Pointer register
0x01c	TRG	RW	0x--------	32-bit	Trigger Counter register
0x020	CTL	RW	0x00000000	32-bit	Control Register
0x024	RWD	WO	0x00000000	32-bit	RAM Write Data register
0x028	MODE	RW	0x000000--	32-bit	Mode register
0x02c	LBUFLEVEL	RO	0x--------	32-bit	Latched Buffer Fill Level
0x030	CBUFLEVEL	RO	0x--------	32-bit	Current Buffer Fill Level
0x034	BUFWM	RW	0x--------	32-bit	Buffer Level Water Mark
0x300	FFSR	RO	0x0000000-	32-bit	Formatter and Flush Status Register
0x304	FFCR	RW	0x00000000	32-bit	Formatter and Flush Control Register
0x308	PSCR	RW	0x0000000A	32-bit	Periodic Synchronization Counter Register
0xed0	ITATBMDATA0	WO	0x00000000	32-bit	Integration Test ATB Transmitter Data 0 register
0xed4	ITATBMCTR2	RO	0x00000000	32-bit	Integration Test ATB Transmitter Control 2 register
0xed8	ITATBMCTR1	WO	0x00000000	32-bit	Integration Test ATB Transmitter Control 1 register
0xedc	ITATBMCTR0	WO	0x00000000	32-bit	Integration Test ATB Transmitter Control 0 register
0xee0	ITEVTINTR	WO	0x00000000	32-bit	Integration Test Event and Interrupt Control Register
0xee8	ITTRFLIN	RO	0x00000000	32-bit	Integration Test Trigger In and Flush In register
0xeec	ITATBDATA0	RO	0x00000000	32-bit	Integration Test ATB Data 0 Register
0xef0	ITATBCTR2	WO	0x00000000	32-bit	Integration Test ATB Control 2 Register
0xef4	ITATBCTR1	RO	0x00000000	32-bit	Integration Test ATB Control 1 Register
0xef8	ITATBCTR0	RO	0x00000000	32-bit	Integration Test ATB Control 0 Register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x0000000F	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x00000000	32-bit	Authentication Status Register
0xfc4	DEVID1	RO	0x00000001	32-bit	Device Configuration Register 1
0xfc8	DEVID	RO	0x00000-80	32-bit	Device Configuration Register
0xfcc	DEVTYPE	RO	0x00000032	32-bit	Device Type Identifier Register

Offset	Name	Type	Reset	Width	Description
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000EA	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000009B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.19.1 css600_tmc_etf RAM Size register, RSZ

The RSZ register defines the size of trace memory in units of 32-bit words.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x004

Type

RO

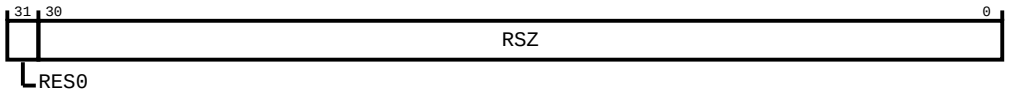
Reset value

0x--------

Bit descriptions

The following figure shows the RSZ register bit assignments.

Figure 9-500: Bit assignment diagram for the RSZ register



The following table shows the RSZ register bit descriptions.

Table 9-518: RSZ bit descriptions

Bits	Name	Reset	Type	Description
31	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
30:0	RSZ	**IMPLEMENTATION DEFINED**	RO	RAM size. Indicates the size of the RAM in 32-bit words. For example: Returns 0x00000100 if trace memory size is 1KB, 0x40000000 if trace memory size is 4GB. This field has the same value as the MEM_SIZE parameter.

9.19.2 css600_tmc_etf Status register, STS

The STS register indicates the status of the Trace Memory Controller. After a reset, software must ignore all the fields of this register except STS.TMCReady. The other fields only have meaning when the TMC has left the Disabled state. Writes to all RO fields of this register are ignored.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x00C

Type

RO

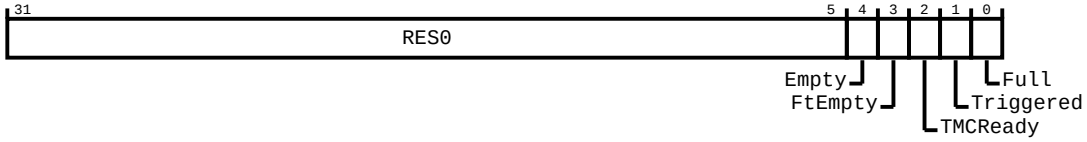
Reset value

0x0000001-

Bit descriptions

The following figure shows the STS register bit assignments.

Figure 9-501: Bit assignment diagram for the STS register



The following table shows the STS register bit descriptions.

Table 9-519: STS bit descriptions

Bits	Name	Reset	Type	Description
31:5	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
4	Empty	0b1	RO	Trace buffer empty. If set, this bit indicates that the trace memory does not contain any valid trace data. The reset value of this bit is 1. On leaving Disabled state, this bit dynamically indicates the empty status of trace memory, CBUFLEVEL == 0. However, this does not mean that the pipeline stages within the TMC are empty. To determine whether the internal pipeline stages are empty, the software must read the STS.TMCReady bit. In Circular Buffer mode the Empty bit and the Full bit in this register can be 1 at the same time. This happens because the Full bit in this mode, when set, does not clear until TraceCaptEn is set.
3	FtEmpty	0b1	RO	Trace capture has been completed and all captured trace data has been written to the trace memory, set in Stopped or Disabled state. Otherwise, it is cleared. The reset value is 1
2	TMCReady	0b1	RO	Trace capture has been completed, all captured trace data has been written to the trace memory, and reading from trace memory completed as a result of FFCR.DrainBuffer bit being set
1	Triggered	UNKNOWN	RO	TMC triggered. This bit is set when trace capture is in progress and the TMC has detected a trigger event. This bit is cleared to 0 when leaving the Disabled state and retains its value when entering the Disabled state. A trigger event is when the TMC has written a set number of data words, as programmed in the TRG register, into the trace memory after a rising edge of trigin input, or a trigger packet (atid_rx = 0x7D) is received in the input trace.
0	Full	UNKNOWN	RO	Trace memory full. This bit helps determine the amount of valid data present in the trace memory. It is not affected by the reprogramming of pointer registers in Disabled state. It is cleared when the TMC leaves Disabled state. In Circular Buffer mode, this flag is set when the RAM write pointer wraps around the top of the buffer. It remains set until the TraceCaptEn bit is cleared and set. In Software FIFO and Hardware FIFO mode, this flag indicates that the current space in the trace memory is less than or equal to the value programmed in the BUFWM Register, that is, Fill level >= MEM_SIZE - BUFWM. The FULL output from the TMC reflects the value of this register bit, except when the Integration Mode bit in the ITCTRL Register, 0xF00, is set.

9.19.3 css600_tmc_etf RAM Read Data register, RRD

Reading this register allows data to be read from the trace memory at the location pointed to by the RRP register when either in the Disabled state or operating in Circular Buffer (CB) or Software FIFO mode 1 (SWF1) mode.

When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the memory width is twice as wide and a memory word holds 8, 16 or 32 bytes. Multiple RRD reads must be performed to read a full memory word. When a full memory width of data has been read via the RRD register, the RRP register is incremented to the next memory word. When the TMC left the Disabled state and the trace memory is empty, this register returns 0xFFFFFFFF. When the TMC left the Disabled state and the trace memory is empty, this register returns 0xFFFFFFFF. When operating in CB mode and the TMC left the Disabled state, this register returns 0xFFFFFFFF in all other states except the Stopped state. When operating in HWF mode, this register also returns 0xFFFFFFFF.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x010

Type

RO

Reset value

0x--------

Bit descriptions

The following figure shows the RRD register bit assignments.

Figure 9-502: Bit assignment diagram for the RRD register

The following table shows the RRD register bit descriptions.

Table 9-520: RRD bit descriptions

Bits	Name	Reset	Type	Description
31:0	RRD	UNKNOWN	RO	Returns the data read from trace memory

9.19.4 css600_tmc_etf RAM Read Pointer register, RRP

The RRP register contains the value of the read pointer that is used to read entries from trace memory over the APB interface. Software must program it with the same value as RWP before enabling trace capture.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x014

Type

RW

Reset value

0x--------

Bit descriptions

The following figure shows the RRP register bit assignments.

Figure 9-503: Bit assignment diagram for the RRP register

The following table shows the RRP register bit descriptions.

Table 9-521: RRP bit descriptions

Bits	Name	Reset	Type	Description
31:0	RRP	**UNKNOWN**	RW	The RRP width depends on the size of trace memory and is given by log2(MEM_SIZE x 4). The remaining MSBs of the 32-bit register are of type **RAZ/WI**. When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the memory width is twice as wide and a memory word holds 8, 16 or 32 bytes. When a full memory width of data has been read via the RRD register, the RRP register is incremented to the next memory word. The lowest 4 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 32 or 64 bits. The lowest 5 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 128 bits.

9.19.5 css600_tmc_etf RAM Write Pointer register, RWP

The RWP register sets the write pointer that writes entries into the trace memory. Software must program it before enabling trace capture.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x018`

Type

RW

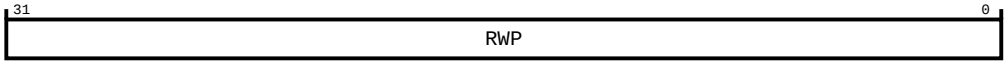
Reset value

`0x--------`

Bit descriptions

The following figure shows the RWP register bit assignments.

Figure 9-504: Bit assignment diagram for the RWP register



The following table shows the RWP register bit descriptions.

Table 9-522: RWP bit descriptions

Bits	Name	Reset	Type	Description
31:0	RWP	UNKNOWN	RW	The RWP width depends on the size of trace memory and is given by log2(MEM_SIZE x 4). The remaining MSBs of the 32-bit register are of type **RAZ/WI**. When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the memory width is twice as wide and a memory word holds 8, 16 or 32 bytes. When a full memory width of data has been written to the RWD register, the RWP register is incremented to the next memory word. The lowest 4 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 32 or 64 bits. The lowest 5 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 128 bits.

9.19.6 css600_tmc_etf Trigger Counter register, TRG

The TRG register, in Circular Buffer mode, specifies the number of 32-bit words to capture in the trace memory, after detecting either a rising edge on the trigin input or a trigger packet in the incoming trace stream, that is, where atid_rx = 0x7D.

The value programmed must be aligned to the frame length of 128 bits. Software must program this register before leaving Disabled state.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x01C`

Type

RW

Reset value

`0x--------`

Bit descriptions

The following figure shows the TRG register bit assignments.

Figure 9-505: Bit assignment diagram for the TRG register

The following table shows the TRG register bit descriptions.

Table 9-523: TRG bit descriptions

Bits	Name	Reset	Type	Description
31:0	TRG	UNKNOWN	RW	Trigger count. This count represents the number of 32-bit words of trace that are captured between a trigger packet and a trigger event. The lowest two bits have access type **RAZ/WI**.

9.19.7 css600_tmc_etf Control Register, CTL

The CTL register controls trace stream capture. Setting the CTL.TraceCaptEn bit to 1 enables the TMC to capture the trace data. When trace capture is enabled, the formatter behavior is controlled by the FFCR register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x020

Type

RW

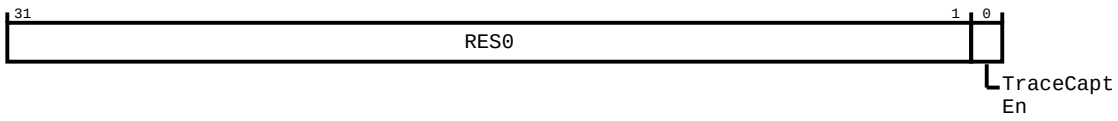
Reset value

0x00000000

Bit descriptions

The following figure shows the CTL register bit assignments.

Figure 9-506: Bit assignment diagram for the CTL register



The following table shows the CTL register bit descriptions.

Table 9-524: CTL bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	TraceCaptEn	0b0	RW	Trace capture enable: **0b0** Disable trace capture **0b1** Enable trace capture

9.19.8 css600_tmc_etf RAM Write Data register, RWD

The RWD register enables testing of trace memory connectivity to the TMC. Writing this register allows data to be written to the trace memory at the location pointed to by the RWP register when in the Disabled state.

When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the memory width is twice as wide and a memory word holds 8, 16 or 32 bytes. Multiple RWD writes must be performed to write a full memory word. When a full memory width of data has been written via the RWD register, the data is written to the trace memory and the RWP register is incremented to the next memory word.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x024`

Type

WO

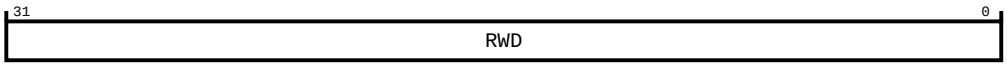
Reset value

`0x00000000`

Bit descriptions

The following figure shows the RWD register bit assignments.

Figure 9-507: Bit assignment diagram for the RWD register



The following table shows the RWD register bit descriptions.

Table 9-525: RWD bit descriptions

Bits	Name	Reset	Type	Description
31:0	RWD	`0x0`	WO	Data written to this register is placed in the trace memory.

9.19.9 css600_tmc_etf Mode register, MODE

The MODE register controls the TMC operating mode. The operating mode can only be changed when the TMC is in Disabled state. Attempting to write to this register in any other state results in **UNPREDICTABLE** behavior. The operating mode is ignored when in Disabled state.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x028`

Type

RW

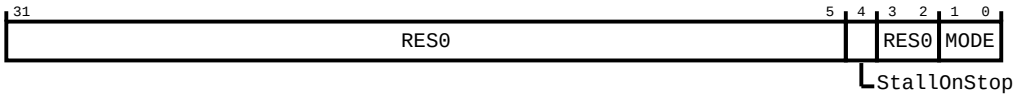
Reset value

`0x000000--`

Bit descriptions

The following figure shows the MODE register bit assignments.

Figure 9-508: Bit assignment diagram for the MODE register



The following table shows the MODE register bit descriptions.

Table 9-526: MODE bit descriptions

Bits	Name	Reset	Type	Description
31:5	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
4	StallOnStop	**UNKNOWN**	RW	Stall On Stop. If this bit is set and the formatter stops as a result of a stop event, the output atready_rx is de-asserted to stall the ATB interface and avoid loss of trace. If this bit is clear and the formatter stops as a result of a stop event, signal atready_rx remains asserted but the TMC discards further incoming trace.
3:2	**RES0**	`0b00`	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
1:0	MODE	UNKNOWN	RW	Selects the operating mode. If a reserved MODE value is programmed and trace capture is enabled, the TMC starts to operate in Software FIFO mode 1 (SWF1) mode. However, reading the MODE.MODE field returns the programmed value. **0b00** CB, Circular Buffer mode. **0b01** SWF1, Software Read FIFO mode 1. **0b10** HWF, Hardware Read FIFO mode. **0b11** Reserved. (SWF1)

9.19.10 css600_tmc_etf Latched Buffer Fill Level, LBUFLEVEL

Reading the LBUFLEVEL register returns the maximum fill level of the trace memory in 32-bit words since this register was last read. Reading this register also results in its contents being updated to the current fill level. When entering Disabled state, it retains its last value.

While in Disabled state, reads from this register do not affect its value. When exiting Disabled state, the LBUFLEVEL register is updated to the current fill level.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x02C

Type

RO

Reset value

0x--------

Bit descriptions

The following figure shows the LBUFLEVEL register bit assignments.

Figure 9-509: Bit assignment diagram for the LBUFLEVEL register

The following table shows the LBUFLEVEL register bit descriptions.

Table 9-527: LBUFLEVEL bit descriptions

Bits	Name	Reset	Type	Description
31:0	LBUFLEVEL	UNKNOWN	RO	Latched Buffer Fill Level. Indicates the maximum fill level of the trace memory in 32-bit words since this register was last read. The width of the register is 1 + log2(MEM_SIZE).

9.19.11 css600_tmc_etf Current Buffer Fill Level, CBUFLEVEL

The CBUFLEVEL register indicates the current fill level of the trace memory in units of 32-bit words. When the TMC leaves Disabled state, this register dynamically indicates the current fill level of trace memory. It retains its value on entering Disabled state.

It is not affected by the reprogramming of pointer registers in Disabled state with the exception of RRD reads and RWD writes. Before leaving the Disabled state software must program RRP with the same value as RWP. Without doing this results in **UNPREDICTABLE** behavior.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x030`

Type

RO

Reset value

`0x--------`

Bit descriptions

The following figure shows the CBUFLEVEL register bit assignments.

Figure 9-510: Bit assignment diagram for the CBUFLEVEL register

The following table shows the CBUFLEVEL register bit descriptions.

Table 9-528: CBUFLEVEL bit descriptions

Bits	Name	Reset	Type	Description
31:0	CBUFLEVEL	UNKNOWN	RO	Current Buffer Fill Level. Indicates the current fill level of the trace memory in 32-bit words. The width of the register is 1 + log2(MEM_SIZE).

9.19.12 css600_tmc_etf Buffer Level Water Mark, BUFWM

The value that is programmed into the BUFWM register indicates the required threshold vacancy level in 32-bit words in the trace memory.

When the available space in the FIFO is less than or equal to this value, that is, fill level >= (MEM_SIZE - BUFWM), the full output is asserted and the STS.Full bit is set. This register is used only in the FIFO modes, that is, Software FIFO mode 1 (SWF1), Software FIFO mode 2 (SWF2), and Hardware FIFO (HWF) modes. In Circular Buffer (CB) mode, the same functionality is obtained by programming the RWP to the required vacancy trigger level, so that when the pointer wraps around, the full output gets asserted indicating that the vacancy level has fallen below the required level. Reading this register returns the programmed value. The maximum value that can be written into this register is MEM_SIZE - 1, in which case the full output is asserted after the first 32-bit word is written to trace memory. Writing to this register other than when in Disabled state results in **UNPREDICTABLE** behavior. Any software using it must program it with an initial value before setting the CTL.TraceCaptEn bit to 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x034`

Type

RW

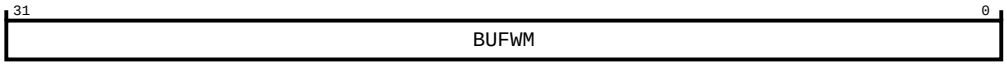
Reset value

`0x--------`

Bit descriptions

The following figure shows the BUFWM register bit assignments.

Figure 9-511: Bit assignment diagram for the BUFWM register



The following table shows the BUFWM register bit descriptions.

Table 9-529: BUFWM bit descriptions

Bits	Name	Reset	Type	Description
31:0	BUFWM	**UNKNOWN**	RW	Buffer Level Watermark. Indicates the desired threshold vacancy level in 32-bit words in the trace memory. The width of the register is log2(MEM_SIZE).

9.19.13 css600_tmc_etf Formatter and Flush Status Register, FFSR

This register indicates the status of the Formatter, and the status of Flush request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x300

Type

RO

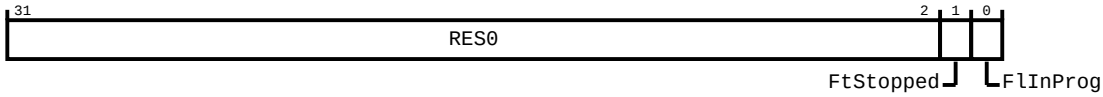
Reset value

0x0000000-

Bit descriptions

The following figure shows the FFSR register bit assignments.

Figure 9-512: Bit assignment diagram for the FFSR register



The following table shows the FFSR register bit descriptions.

Table 9-530: FFSR bit descriptions

Bits	Name	Reset	Type	Description
31:2	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
1	FtStopped	UNKNOWN	RO	Formatter Stopped. This bit behaves the same way as STS.FtEmpty.
0	FlInProg	UNKNOWN	RO	Flush In Progress. This bit indicates whether the TMC is currently processing a flush request. The flush initiation is controlled by the flush control bits in the FFCR register. The flush request could additionally be from the ATB transmitter port. This bit is cleared to 0 when leaving the Disabled state and retains its value when entering the Disabled state. When in Disabled state, this bit is not updated. **0b0** No flush activity in progress. **0b1** Flush in progress on the ATB receiver interface or the TMC internal pipeline.

9.19.14 css600_tmc_etf Formatter and Flush Control Register, FFCR

The FFCR controls the generation of stop, trigger and flush events. The insertion of a flush completion packet and the insertion of a trigger packet in the formatted trace is enabled here.

Also one of the 2 formatter modes for bypass mode and normal mode can be changed here when the formatter has stopped.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x304`

Type

RW

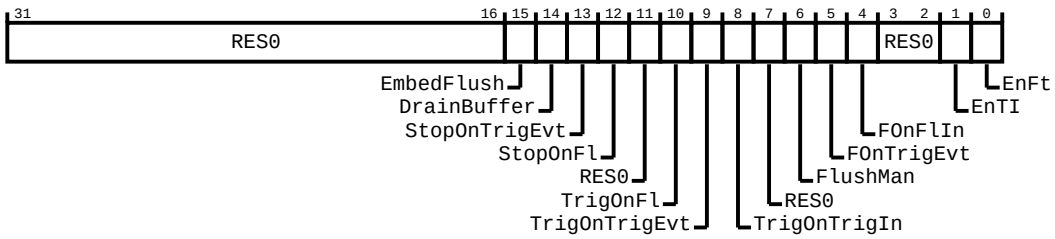
Reset value

`0x00000000`

Bit descriptions

The following figure shows the FFCR register bit assignments.

Figure 9-513: Bit assignment diagram for the FFCR register



The following table shows the FFCR register bit descriptions.

Table 9-531: FFCR bit descriptions

Bits	Name	Reset	Type	Description
31:16	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15	EmbedFlush	0b0	RW	Embed Flush ID (flush completion packet). Enables insertion of Flush ID 0x7B with a single byte of data payload = 0x00 in the output trace, immediately after the last flush data byte, when a flush completes on the ATB receiver interface. This bit is effective only in Normal formatting modes. In Bypass mode, the Flush ID insertion remains disabled and this bit is ignored. **0b0** Disable Flush ID insertion. **0b1** Enable Flush ID insertion.
14	DrainBuffer	0b0	WO	Drain Buffer. This bit is used to enable draining of the trace data through the ATB transmitter interface after the formatter has stopped. It is useful in CB mode to capture trace data into trace memory and then to drain the captured trace through the ATB transmitter interface. Writing a 1 to this bit when in Stopped state starts the drain of the contents of trace buffer. This bit always reads as 0. The STS.TMCReady bit goes LOW while the drain is in progress. This bit is functional only when the TMC is in CB mode and formatting is enabled, that is, FFCR.EnFt=1. Setting this bit when the TMC is in any other mode, or when not in Stopped state, results in **UNPREDICTABLE** behavior. When trace capture is complete in CB mode, all of the captured trace must be retrieved from the trace memory, either by reading all trace data out through RRD reads, or draining all trace data by setting the FFCR.DrainBuffer bit. Setting this bit after some of the captured trace has been read out through RRD results in **UNPREDICTABLE** behavior.
13	StopOnTrigEvt	0b0	RW	Stop On Trigger Event. If this bit is set, the formatter is stopped when a Trigger Event has been observed. This bit must be used only in CB mode. If trace capture is enabled in Software FIFO mode 1 (SWF1), or Hardware FIFO (HWF) mode with this bit set, it results in **UNPREDICTABLE** behavior.
12	StopOnFl	0b0	RW	Stop On Flush. If this bit is set, the formatter is stopped on completion of a flush operation. The initiation of a flush operation is controlled by programming the register bits FFCR.FlushMan, FFCR.FOnTrigEvt, and FFCR.FOnFlIn. When a flush-initiation condition occurs, afvalid_rx is asserted, and when the flush completion is received, that is, afready_rx=1, trace capture is stopped. Any remaining data in the formatter is appended with a post-amble and written to trace memory. The flush operation is then complete. If a flush is initiated by the ATB Transmitter interface, its completion does not lead to a formatter stop regardless of the value that is programmed in this bit.
11	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
10	TrigOnFl	0b0	RW	Indicate on trace stream the completion of flush. If this bit is set, a trigger is indicated on the trace stream when afready_rx is received for a flush in progress. If this bit is clear, no triggers are embedded in the trace stream on flush completion. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit. If a flush is initiated by the ATB Transmitter interface, its completion does not lead to a trigger indication on the trace stream regardless of the value that is programmed in this bit.
9	TrigOnTrigEvt	0b0	RW	Indicate on trace stream the occurrence of a Trigger Event. If this bit is set, a trigger is indicated on the output trace stream when a Trigger Event occurs. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit. This bit must be used only in CB mode. If trace capture is enabled in SWF1, or HWF mode with this bit set, it results in **UNPREDICTABLE** behavior.
8	TrigOnTrigIn	0b0	RW	Indicate on trace stream the occurrence of a rising edge on trigin. If this bit is set, a trigger is indicated on the trace stream when a rising edge is detected on the trigin input. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit.
7	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
6	FlushMan	0b0	RW	Manually generate a flush of the system. Writing 1 to this bit causes a flush to be generated. This bit is cleared automatically when, in formatter bypass mode, afready_rx was sampled high, or, in normal formatting mode, afready_rx was sampled high and all flush data was output to the trace memory. If CTL.TraceCaptEn=0, writes to this bit are ignored.

Bits	Name	Reset	Type	Description
5	FOnTrigEvt	0b0	RW	Flush on Trigger Event. If FFCR.StopOnTrigEvt is set, this bit is ignored. Setting this bit generates a flush when a Trigger Event occurs. If FFCR.StopOnTrigEvt is set, this bit is ignored. This bit must be used only in CB mode. If trace capture is enabled in SWF1, or HWF mode with this bit set, it results in **UNPREDICTABLE** behavior.
4	FOnFlIn	0b0	RW	Setting this bit enables the detection of transitions on the flushin input by the TMC. If this bit is set and the formatter has not already stopped, a rising edge on flushin initiates a flush request.
3:2	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
1	EnTI	0b0	RW	Enable Trigger Insertion. Setting this bit enables the insertion of triggers in the formatted trace stream. A trigger is indicated by inserting one byte of data 0x00 with atid_rx=0x7D in the trace stream. Trigger indication on the trace stream is also controlled by the register bits FFCR.TrigOnFl, FFCR.TrigOnTrigEvt, and FFCR.TrigOnTrigIn. This bit can only be changed when the TMC is in Disabled state. If FFCR.EnTI bit is set formatting is enabled.
0	EnFt	0b0	RW	Enable Formatter. If this bit is set, formatting is enabled. When EnTi is set, formatting is enabled. When CB mode is not used, formatting is also enabled. For backwards-compatibility with earlier versions of the ETB disabling of formatting is supported only in CB mode. This bit can only be changed when TMC is in Disabled state.

9.19.15 css600_tmc_etf Periodic Synchronization Counter Register, PSCR

This register determines the reload value of the Periodic Synchronization Counter. This counter enables the frequency of sync packets to be optimized to the trace capture buffer size. The default behavior of the counter is to generate periodic synchronization requests, syncreq_rx, on the ATB receiver interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x308

Type

RW

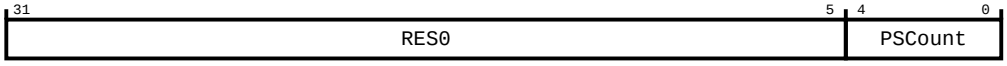
Reset value

0x0000000A

Bit descriptions

The following figure shows the PSCR register bit assignments.

Figure 9-514: Bit assignment diagram for the PSCR register



The following table shows the PSCR register bit descriptions.

Table 9-532: PSCR bit descriptions

Bits	Name	Reset	Type	Description
31:5	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
4:0	PSCount	0b01010	RW	Periodic Synchronization Count. Determines the reload value of the Synchronization Counter. The reload value takes effect the next time the counter reaches zero. When trace capture is enabled, the Synchronization Counter counts the number of bytes of trace data that is stored into the trace memory, regardless of whether the trace data has been formatted by the TMC or not, since the occurrence of the last sync request on the ATB receiver interface. When the counter reaches 0, a sync request is sent on the ATB receiver interface. Reads from this register return the reload value that is programmed in this register. This field resets to 0x0A, that is, the default sync period is 2^10 bytes. If a reserved value is programmed in this register field, the value 0x1B is used instead, and subsequent reads from this register also return 0x1B. The following constraints apply to the values written to the PSCount field: 0x0 - synchronization is disabled, 0x1-0x6 - reserved, 0x7-0x1B - synchronization period is 2^PSCount bytes. The smallest value 0x7 gives a sync period of 128 bytes. The maximum allowed value 0x1B gives a sync period of 2^27 bytes, 0x1C-0x1F - reserved.

9.19.16 css600_tmc_etf Integration Test ATB Transmitter Data 0 register, ITATBMDATA0

This register enables control of the atdata_tx output in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, the value that is written to any given bit is driven on the output pin that is controlled by that bit and the reads return 0x0.

The width of this register is given by 1+(ATB DATA WIDTH)/8.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xED0

Type

WO

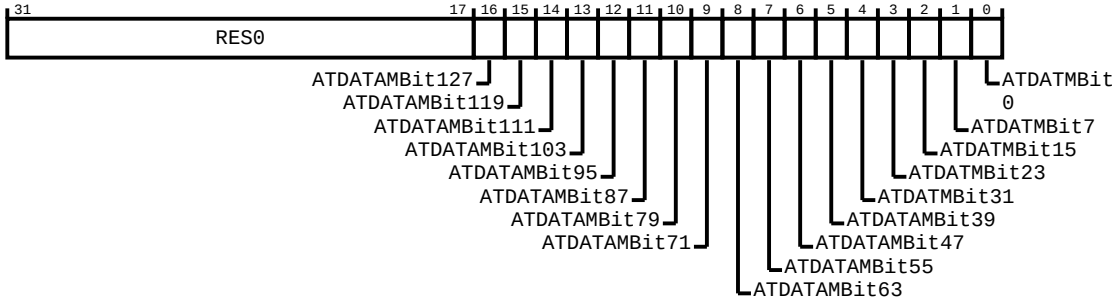
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBMDATA0 register bit assignments.

Figure 9-515: Bit assignment diagram for the ITATBMDATA0 register



The following table shows the ITATBMDATA0 register bit descriptions.

Table 9-533: ITATBMDATA0 bit descriptions

Bits	Name	Reset	Type	Description
31:17	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
16	ATDATAMBit127	0b0	WO	Controls the value of atdata_tx[127] output in integration mode.
15	ATDATAMBit119	0b0	WO	Controls the value of atdata_tx[119] output in integration mode.
14	ATDATAMBit111	0b0	WO	Controls the value of atdata_tx[111] output in integration mode.
13	ATDATAMBit103	0b0	WO	Controls the value of atdata_tx[103] output in integration mode.
12	ATDATAMBit95	0b0	WO	Controls the value of atdata_tx[95] output in integration mode.
11	ATDATAMBit87	0b0	WO	Controls the value of atdata_tx[87] output in integration mode.
10	ATDATAMBit79	0b0	WO	Controls the value of atdata_tx[79] output in integration mode.
9	ATDATAMBit71	0b0	WO	Controls the value of atdata_tx[71] output in integration mode.
8	ATDATAMBit63	0b0	WO	Controls the value of atdata_tx[63] output in integration mode.
7	ATDATAMBit55	0b0	WO	Controls the value of atdata_tx[55] output in integration mode.
6	ATDATAMBit47	0b0	WO	Controls the value of atdata_tx[47] output in integration mode.
5	ATDATAMBit39	0b0	WO	Controls the value of atdata_tx[39] output in integration mode.
4	ATDATMBit31	0b0	WO	Controls the value of atdata_tx[31] output in integration mode.
3	ATDATMBit23	0b0	WO	Controls the value of atdata_tx[23] output in integration mode.
2	ATDATMBit15	0b0	WO	Controls the value of atdata_tx[15] output in integration mode.
1	ATDATMBit7	0b0	WO	Controls the value of atdata_tx[7] output in integration mode.
0	ATDATMBit0	0b0	WO	Controls the value of atdata_tx[0] output in integration mode.

9.19.17 css600_tmc_etf Integration Test ATB Transmitter Control 2 register, ITATBMCTR2

This register captures the values of ATB transmitter inputs atready_tx, afvalid_tx, and syncreq_tx in integration mode. In functional mode, this register behaves as **RAZ/WI**.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xED4

Type

RO

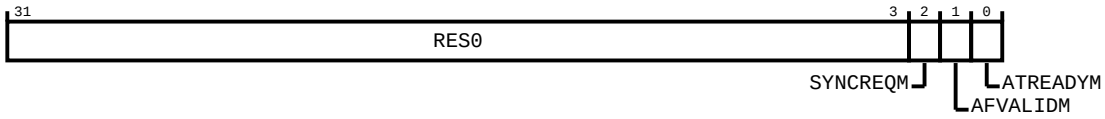
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBMCTR2 register bit assignments.

Figure 9-516: Bit assignment diagram for the ITATBMCTR2 register



The following table shows the ITATBMCTR2 register bit descriptions.

Table 9-534: ITATBMCTR2 bit descriptions

Bits	Name	Reset	Type	Description
31:3	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
2	SYNCREQM	0b0	RO	Integration status of syncreq_tx input. In integration mode, this bit latches to 1 on a rising edge of syncreq_tx input, which is cleared when this register is read or when integration mode is disabled.
1	AFVALIDM	0b0	RO	Integration status of afvalid_tx input. In integration mode, writes are ignored and reads return the value of afvalid_tx input.
0	ATREADYM	0b0	RO	Integration status of atready_tx input. In integration mode, writes are ignored and reads return the value of the atready_tx input.

9.19.18 css600_tmc_etf Integration Test ATB Transmitter Control 1 register, ITATBMCTR1

This register enables control of the atid_tx output in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, the value that is written to this register is driven on the atid_tx output and the reads return 0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xED8

Type

WO

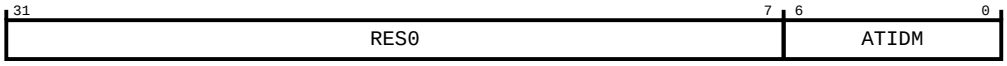
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBMCTR1 register bit assignments.

Figure 9-517: Bit assignment diagram for the ITATBMCTR1 register



The following table shows the ITATBMCTR1 register bit descriptions.

Table 9-535: ITATBMCTR1 bit descriptions

Bits	Name	Reset	Type	Description
31:7	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
6:0	ATIDM	0x0	WO	Controls the value of the atid_tx[6:0] output in integration mode.

9.19.19 css600_tmc_etf Integration Test ATB Transmitter Control 0 register, ITATBMCTR0

This register enables control of the ATB transmitter outputs atbytes_tx, atwakeup_tx, afready_tx, and atvalid_tx in integration mode. In functional mode, this register behaves as **RAZ/WI**.

In integration mode, the value that is written to any bit of this register is driven on the output pin that is controlled by that bit, and the reads return 0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEDC

Type

WO

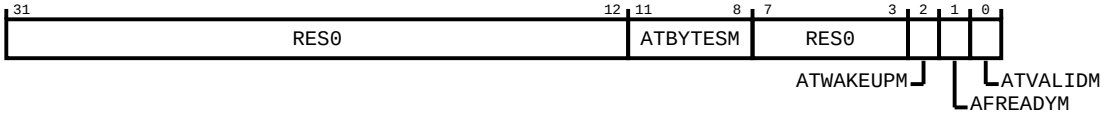
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBMCTR0 register bit assignments.

Figure 9-518: Bit assignment diagram for the ITATBMCTR0 register



The following table shows the ITATBMCTR0 register bit descriptions.

Table 9-536: ITATBMCTR0 bit descriptions

Bits	Name	Reset	Type	Description
31:12	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
11:8	ATBYTESM	0b0000	WO	Controls the value of atbytes_tx output in integration mode. This width of this field is log2(ATB DATA WIDTH/8).
7:3	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
2	ATWAKEUPM	0b0	WO	Controls the value of atwakeup_tx output in integration mode.
1	AFREADYM	0b0	WO	Controls the value of afready_tx output in integration mode.
0	ATVALIDM	0b0	WO	Controls the value of atvalid_tx output in integration mode.

9.19.20 css600_tmc_etf Integration Test Event and Interrupt Control Register, ITEVTINTR

This register controls the values of event and interrupt outputs in integration mode. In functional mode, this register behaves as **RAZ/WI**.

In integration mode, the value that is written to any bit of this register is driven on the output pin that is controlled by that bit and the reads return 0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEE0

Type

WO

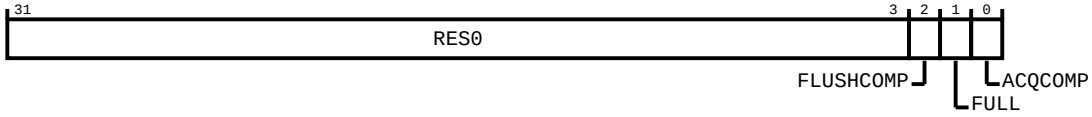
Reset value

0x00000000

Bit descriptions

The following figure shows the ITEVTINTR register bit assignments.

Figure 9-519: Bit assignment diagram for the ITEVTINTR register



The following table shows the ITEVTINTR register bit descriptions.

Table 9-537: ITEVTINTR bit descriptions

Bits	Name	Reset	Type	Description
31:3	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
2	FLUSHCOMP	0b0	WO	Controls the value of flushcomp output in integration mode.
1	FULL	0b0	WO	Controls the value of full output in integration mode.
0	ACQCOMP	0b0	WO	Controls the value of acqcomp output in integration mode.

9.19.21 css600_tmc_etf Integration Test Trigger In and Flush In register, ITTRFLIN

This register captures the values of the flushin and trigin inputs in integration mode. In functional mode, this register behaves as **RAZ/WI**.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEE8

Type

RO

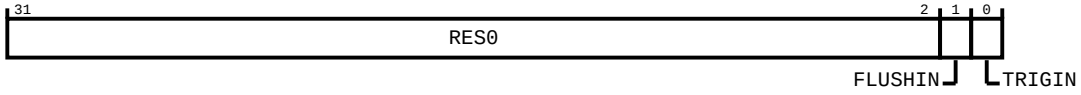
Reset value

0x00000000

Bit descriptions

The following figure shows the ITTRFLIN register bit assignments.

Figure 9-520: Bit assignment diagram for the ITTRFLIN register



The following table shows the ITTRFLIN register bit descriptions.

Table 9-538: ITTRFLIN bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	FLUSHIN	0b0	RO	Integration status of flushin input. In integration mode, this bit latches to 1 on a rising edge of the flushin input. It is cleared when the register is read or when integration mode is disabled.
0	TRIGIN	0b0	RO	Integration status of trigin input. In integration mode, this bit latches to 1 on a rising edge of the trigin input. It is cleared when the register is read or when integration mode is disabled.

9.19.22 css600_tmc_etf Integration Test ATB Data 0 Register, ITATBDATA0

This register captures the value of atdata_rx input in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of corresponding atdata_rx bits. The width of this register is given by: 1+(ATB DATA WIDTH)/8.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEEC

Type

RO

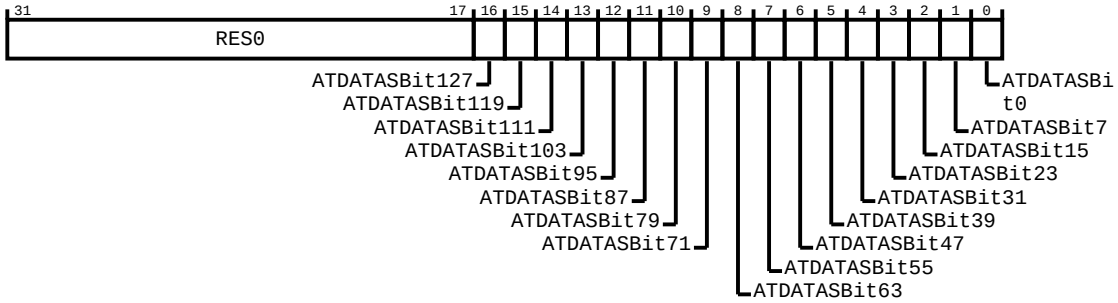
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBDATA0 register bit assignments.

Figure 9-521: Bit assignment diagram for the ITATBDATA0 register



The following table shows the ITATBDATA0 register bit descriptions.

Table 9-539: ITATBDATA0 bit descriptions

Bits	Name	Reset	Type	Description
31:17	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
16	ATDATASBit127	0b0	RO	Returns the value of atdata_rx[127] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
15	ATDATASBit119	0b0	RO	Returns the value of atdata_rx[119] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
14	ATDATASBit111	0b0	RO	Returns the value of atdata_rx[111] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
13	ATDATASBit103	0b0	RO	Returns the value of atdata_rx[103] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
12	ATDATASBit95	0b0	RO	Returns the value of atdata_rx[95] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
11	ATDATASBit87	0b0	RO	Returns the value of atdata_rx[87] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
10	ATDATASBit79	0b0	RO	Returns the value of atdata_rx[79] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
9	ATDATASBit71	0b0	RO	Returns the value of atdata_rx[71] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
8	ATDATASBit63	0b0	RO	Returns the value of atdata_rx[63] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
7	ATDATASBit55	0b0	RO	Returns the value of atdata_rx[55] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
6	ATDATASBit47	0b0	RO	Returns the value of atdata_rx[47] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
5	ATDATASBit39	0b0	RO	Returns the value of atdata_rx[39] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
4	ATDATASBit31	0b0	RO	Returns the value of atdata_rx[31] input in integration mode.
3	ATDATASBit23	0b0	RO	Returns the value of atdata_rx[23] input in integration mode.
2	ATDATASBit15	0b0	RO	Returns the value of atdata_rx[15] input in integration mode.
1	ATDATASBit7	0b0	RO	Returns the value of atdata_rx[7] input in integration mode.
0	ATDATASBit0	0b0	RO	Returns the value of atdata_rx[0] input in integration mode.

9.19.23 css600_tmc_etf Integration Test ATB Control 2 Register, ITATBCTR2

This register enables control of ATB receiver outputs atready_rx, afvalid_rx, and syncreq_rx in integration mode. In functional mode, this register behaves as **RAZ/WI**.

In integration mode, the value that is written to any bit of this register is driven on the output pin that is controlled by that bit and the reads return 0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEF0`

Type

WO

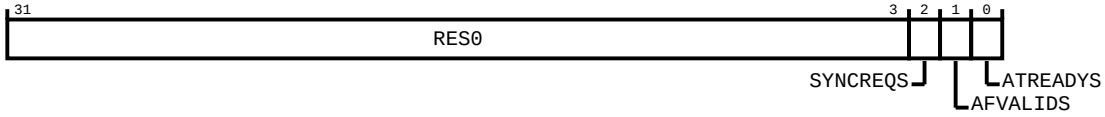
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITATBCTR2 register bit assignments.

Figure 9-522: Bit assignment diagram for the ITATBCTR2 register



The following table shows the ITATBCTR2 register bit descriptions.

Table 9-540: ITATBCTR2 bit descriptions

Bits	Name	Reset	Type	Description
31:3	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
2	SYNCREQS	`0b0`	WO	Controls the value of syncreq_rx output in integration mode.
1	AFVALIDS	`0b0`	WO	Controls the value of afvalid_rx output in integration mode.
0	ATREADYS	`0b0`	WO	Controls the value of atready_rx output in integration mode.

9.19.24 css600_tmc_etf Integration Test ATB Control 1 Register, ITATBCTR1

This register captures the value of the atid_rx[6:0] input in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of atid_rx input.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF4

Type

RO

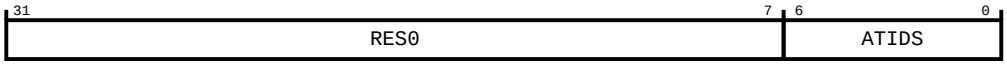
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR1 register bit assignments.

Figure 9-523: Bit assignment diagram for the ITATBCTR1 register



The following table shows the ITATBCTR1 register bit descriptions.

Table 9-541: ITATBCTR1 bit descriptions

Bits	Name	Reset	Type	Description
31:7	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
6:0	ATIDS	0x0	RO	Returns the value of atid_rx[6:0] input in integration mode.

9.19.25 css600_tmc_etf Integration Test ATB Control 0 Register, ITATBCTR0

This register captures the values of ATB receiver inputs atvalid_rx, afready_rx, atwakeup_rx, and atbytes_rx in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of corresponding input pins.

The width of this register is given by: 8+log2(ATB DATA WIDTH/8).

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF8

Type

RO

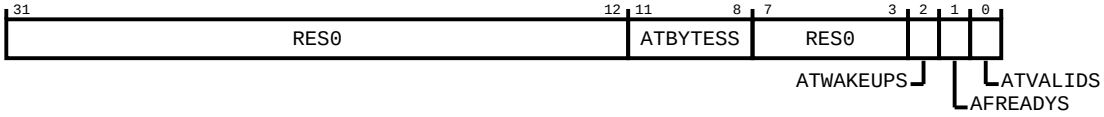
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR0 register bit assignments.

Figure 9-524: Bit assignment diagram for the ITATBCTR0 register



The following table shows the ITATBCTR0 register bit descriptions.

Table 9-542: ITATBCTR0 bit descriptions

Bits	Name	Reset	Type	Description
31:12	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
11:8	ATBYTESS	0b0000	RO	Returns the value of atbytes_rx input in integration mode. N=8+log2(ATB DATA WIDTH/8).
7:3	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
2	ATWAKEUPS	0b0	RO	Returns the value of atwakeup_rx input in integration mode.
1	AFREADYS	0b0	RO	Returns the value of afready_rx input in integration mode.
0	ATVALIDS	0b0	RO	Returns the value of atvalid_rx input in integration mode.

9.19.26 css600_tmc_etf Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

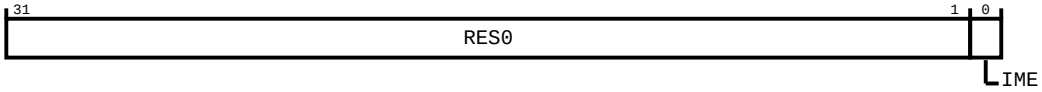
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-525: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-543: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.19.27 css600_tmc_etf Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

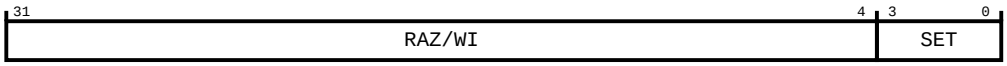
Reset value

0x0000000F

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-526: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-544: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	0x0	RO	**RAZ/WI.**
3:0	SET	0b1111	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.19.28 css600_tmc_etf Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA4

Type

RW

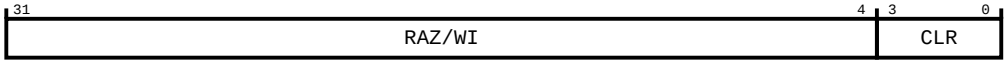
Reset value

0x00000000

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-527: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-545: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	0x0	RO	**RAZ/WI.**
3:0	CLR	0b0000	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.19.29 css600_tmc_etf Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFB8`

Type

RO

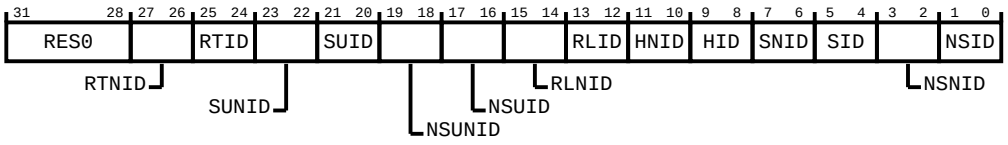
Reset value

`0x00000000`

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-528: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-546: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	`0b0000`	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	`0b00`	RO	Root non-invasive debug. **`0b00`** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.
25:24	RTID	`0b00`	RO	Root invasive debug. **`0b00`** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.
23:22	SUNID	`0b00`	RO	Secure unprivileged non-invasive debug. **`0b00`** Debug level is not supported.

Bits	Name	Reset	Type	Description
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	0b00	RO	Secure non-invasive debug. **0b00** Debug level is not supported.
5:4	SID	0b00	RO	Secure invasive debug. **0b00** Debug level is not supported.
3:2	NSNID	0b00	RO	Non-secure non-invasive debug. **0b00** Debug level is not supported.
1:0	NSID	0b00	RO	Non-secure invasive debug. **0b00** Debug level is not supported.

9.19.30 css600_tmc_etf Device Configuration Register 1, DEVID1

Contains an **IMPLEMENTATION DEFINED** value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC4

Type

RO

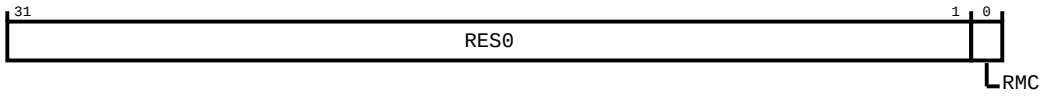
Reset value

0x00000001

Bit descriptions

The following figure shows the DEVID1 register bit assignments.

Figure 9-529: Bit assignment diagram for the DEVID1 register



The following table shows the DEVID1 register bit descriptions.

Table 9-547: DEVID1 bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	RMC	0b1	RO	Register management mode. TMC implements register management mode 1.

9.19.31 css600_tmc_etf Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC8

Type

RO

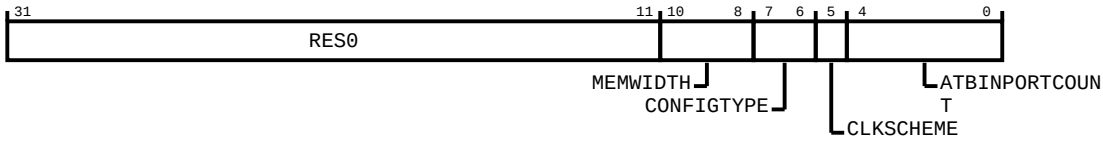
Reset value

0x00000-80

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-530: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-548: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:11	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
10:8	MEMWIDTH	**IMPLEMENTATION DEFINED**	RO	Indicates the width of the internal memory data bus. For the ETF this value is twice ATB_DATA_WIDTH. **0b011** Memory interface databus is 64 bits wide. (ATB_DATA_WIDTH = 32bit) **0b100** Memory interface databus is 128 bits wide. (ATB_DATA_WIDTH = 64bit) **0b101** Memory interface databus is 256 bits wide. (ATB_DATA_WIDTH = 128bit)
7:6	CONFIGTYPE	0b10	RO	Indicates the TMC configuration. **0b10** ETF - Embedded Trace FIFO
5	CLKSCHEME	0b0	RO	RAM Clocking Scheme. This value indicates the TMC RAM clocking scheme used, that is, whether the TMC RAM operates synchronously or asynchronously to the TMC clock. Fixed to 0 indicating that TMC RAM clock is synchronous to the clk input.
4:0	ATBINPORTCOUNT	0b00000	RO	Hidden Level of ATB input multiplexing. This value indicates the type/number of ATB multiplexing present on the input ATB. Fixed to 0x00 indicating that no multiplexing is present.

9.19.32 css600_tmc_etf Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

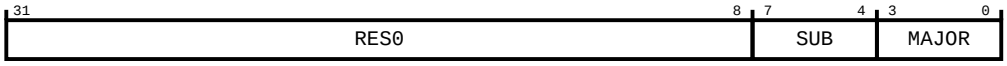
Reset value

0x00000032

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-531: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-549: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	0b0011	RO	Minor classification. Returns 0x3, indicating this component is a FIFO.
3:0	MAJOR	0b0010	RO	Major classification. Returns 0x2, indicating this component is a Trace Link.

9.19.33 css600_tmc_etf Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

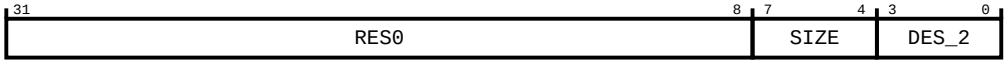
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-532: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-550: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.19.34 css600_tmc_etf Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

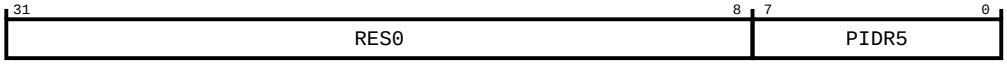
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-533: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-551: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.19.35 css600_tmc_etf Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD8

Type

RO

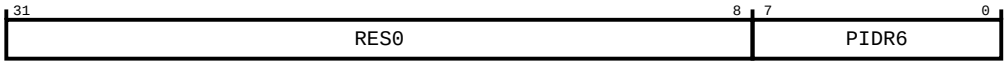
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-534: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-552: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.19.36 css600_tmc_etf Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

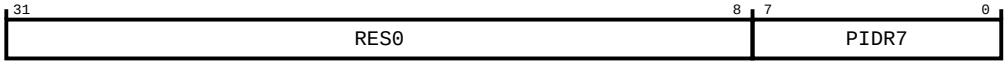
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-535: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-553: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.19.37 css600_tmc_etf Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

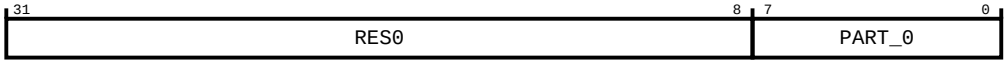
Reset value

0x000000EA

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-536: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-554: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xEA	RO	Part number (lower 8 bits). **0xEA** ETF - Embedded Trace FIFO

9.19.38 css600_tmc_etf Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

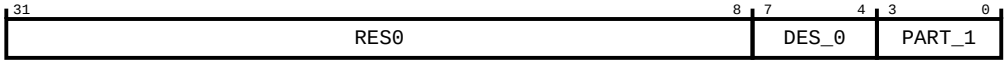
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-537: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-555: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.19.39 css600_tmc_etf Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

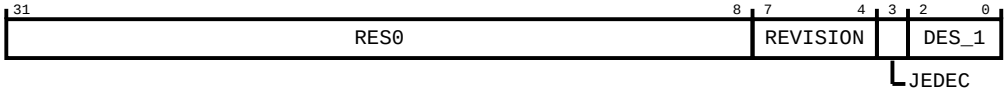
Reset value

0x0000009B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-538: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-556: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b1001	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.

Bits	Name	Reset	Type	Description
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.19.40 css600_tmc_etf Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFEC

Type

RO

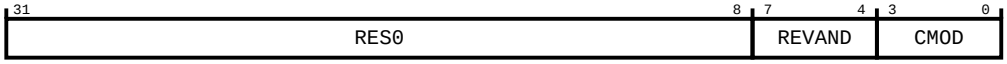
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-539: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-557: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.19.41 css600_tmc_etf Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF0`

Type

RO

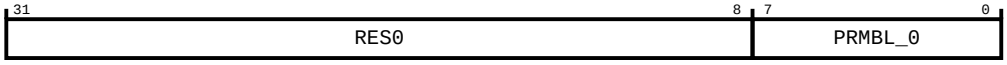
Reset value

`0x0000000D`

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-540: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-558: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	`0xD`	RO	Preamble. Returns `0x0D`.

9.19.42 css600_tmc_etf Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF4`

Type

RO

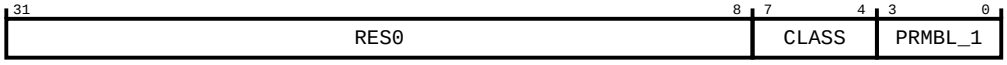
Reset value

`0x00000090`

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-541: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-559: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	`0b1001`	RO	Component class **`0b1001`** CoreSight component
3:0	PRMBL_1	`0b0000`	RO	Preamble. Returns `0x0`.

9.19.43 css600_tmc_etf Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF8`

Type

RO

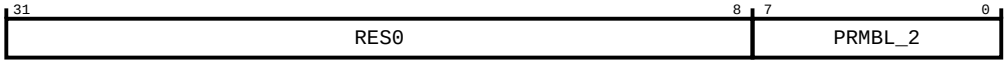
Reset value

`0x00000005`

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-542: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-560: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.19.44 css600_tmc_etf Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

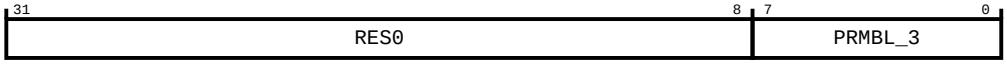
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-543: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-561: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.20 css600_tmc_etr register summary

This section describes the css600_tmc_etr_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-562: css600_tmc_etr_registers register summary

Offset	Name	Type	Reset	Width	Description
0x004	RSZ	RW	0x--------	32-bit	RAM Size register
0x00c	STS	RW	0x000000--	32-bit	Status register
0x010	RRD	RO	0x--------	32-bit	RAM Read Data register
0x014	RRP	RW	0x--------	32-bit	RAM Read Pointer register
0x018	RWP	RW	0x--------	32-bit	RAM Write Pointer register
0x01c	TRG	RW	0x--------	32-bit	Trigger Counter register
0x020	CTL	RW	0x00000000	32-bit	Control Register
0x024	RWD	WO	0x00000000	32-bit	RAM Write Data register
0x028	MODE	RW	0x000000--	32-bit	Mode register
0x02c	LBUFLEVEL	RO	0x--------	32-bit	Latched Buffer Fill Level
0x030	CBUFLEVEL	RO	0x--------	32-bit	Current Buffer Fill Level
0x034	BUFWM	RW	0x--------	32-bit	Buffer Level Water Mark
0x038	RRPHI	RW	0x--------	32-bit	RAM Read Pointer High register
0x03c	RWPHI	RW	0x--------	32-bit	RAM Write Pointer High register
0x110	AXICTL	RW	0x00000-0-	32-bit	AXI Control Register
0x118	DBALO	RW	0x--------	32-bit	Data Buffer Address Low register
0x11c	DBAHI	RW	0x00000000	32-bit	Data Buffer Address HIGH register
0x120	RURP	WO	0x00000000	32-bit	RAM Update Read Pointer register
0x300	FFSR	RO	0x0000000-	32-bit	Formatter and Flush Status Register
0x304	FFCR	RW	0x00000000	32-bit	Formatter and Flush Control Register
0x308	PSCR	RW	0x0000000A	32-bit	Periodic Synchronization Counter Register
0xed0	AXICTL1	RW	0x0---00--	32-bit	AXI Control Register 1
0xee0	ITEVTINTR	WO	0x00000000	32-bit	Integration Test Event and Interrupt Control Register
0xee8	ITTRFLIN	RO	0x00000000	32-bit	Integration Test Trigger In and Flush In register
0xeec	ITATBDATA0	RO	0x00000000	32-bit	Integration Test ATB Data 0 Register
0xef0	ITATBCTR2	WO	0x00000000	32-bit	Integration Test ATB Control 2 Register
0xef4	ITATBCTR1	RO	0x00000000	32-bit	Integration Test ATB Control 1 Register
0xef8	ITATBCTR0	RO	0x00000000	32-bit	Integration Test ATB Control 0 Register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x0000000F	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x000000--	32-bit	Authentication Status Register

Offset	Name	Type	Reset	Width	Description
0xfc4	DEVID1	RO	0x00000001	32-bit	Device Configuration Register 1
0xfc8	DEVID	RO	0x03----40	32-bit	Device Configuration Register
0xfcc	DEVTYPE	RO	0x00000021	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000E8	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000009B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.20.1 css600_tmc_etr RAM Size register, RSZ

The RSZ register defines the size of trace memory in units of 32-bit words.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x004

Type

RW

Reset value

0x--------

Bit descriptions

The following figure shows the RSZ register bit assignments.

Figure 9-544: Bit assignment diagram for the RSZ register

The following table shows the RSZ register bit descriptions.

Table 9-563: RSZ bit descriptions

Bits	Name	Reset	Type	Description
31	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
30:0	RSZ	**UNKNOWN**	RW	RAM size. Indicates the size of trace memory in 32-bit words. The size of the trace buffer must be a multiple of the AXI data width. This can be found by looking at the MEMWIDTH field in the DEVID Register. The maximum trace buffer size permitted is 4GB. The minimum trace buffer size enabled in Software FIFO mode and Hardware FIFO mode is 512 bytes. The minimum trace buffer size enabled in Circular Buffer mode is one AXI dataword. Returns the programmed value on reading. The burst length programmed in the AXICTL Register, WrBurstLen, must be compatible with the trace buffer size and the AXI data width so that the total number of bytes of data transferred in a burst is not greater than the trace buffer size. Programming an incompatible burst length results in **UNPREDICTABLE** behavior. Modifying this register when the TMCReady bit, STS Register, 0x00C is clear, or the TraceCaptEn bit, CTL Register, 0x020 is set, results in **UNPREDICTABLE** behavior. Arm recommends that DBA +RSZ*4<2^AXI_ADDR_WIDTH.

9.20.2 css600_tmc_etr Status register, STS

The STS register indicates the status of the Trace Memory Controller. After a reset, software must ignore all the fields of this register except STS.TMCReady. The other fields only have meaning when the TMC has left the Disabled state. Writes to all RO fields of this register are ignored.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x00C

Type

RW

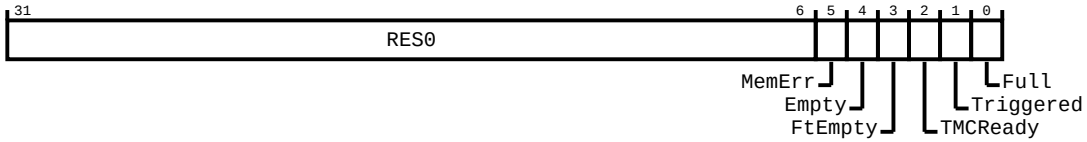
Reset value

0x000000--

Bit descriptions

The following figure shows the STS register bit assignments.

Figure 9-545: Bit assignment diagram for the STS register



The following table shows the STS register bit descriptions.

Table 9-564: STS bit descriptions

Bits	Name	Reset	Type	Description
31:6	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
5	MemErr	**UNKNOWN**	RW	Memory error status. This bit is set when an error has occurred on the AXI manager interface. The error can be caused by either an error response received from the connected AXI subordinate or attempted AXI transfers without proper authentication and Leaving the Disabled state and RWP does not point inside the buffer space (starting with DBA and ending before DBA+RSZ*4). Writes to this bit are allowed in Disabled or Stopped state. Writing 0 has no affect. Writing 1 clears this bit.
4	Empty	0b1	RO	Trace buffer empty. If set, this bit indicates that the trace memory does not contain any valid trace data. The reset value of this bit is 1. On leaving Disabled state, this bit dynamically indicates the empty status of trace memory, CBUFLEVEL == 0. However, this does not mean that the pipeline stages within the TMC are empty. To determine whether the internal pipeline stages are empty, the software must read the STS.TMCReady bit. In Circular Buffer mode the Empty bit and the Full bit in this register can be 1 at the same time. This happens because the Full bit in this mode, when set, does not clear until TraceCaptEn is set.
3	FtEmpty	0b1	RO	Trace capture has been completed and all captured trace data has been written to the trace memory, set in Stopped, Disabled or Draining state. Otherwise, it is cleared. The reset value is 1
2	TMCReady	0b0	RO	Trace capture has been completed, all captured trace data has been written to the trace memory, and reading from trace memory completed as a result of final AXI write completing
1	Triggered	**UNKNOWN**	RO	TMC triggered. This bit is set when trace capture is in progress and the TMC has detected a trigger event. This bit is cleared to 0 when leaving the Disabled state and retains its value when entering the Disabled state. A trigger event is when the TMC has written a set number of data words, as programmed in the TRG register, into the trace memory after a rising edge of trigin input, or a trigger packet (atid_rx = 0x7D) is received in the input trace.
0	Full	**UNKNOWN**	RW	Trace memory full. This bit helps determine the amount of valid data present in the trace memory. In Circular Buffer mode writes to this bit are allowed in Disabled state, for example for clearing this bit. However, it is not affected by the reprogramming of pointer registers in this state. In Circular Buffer mode, this flag is set when the RAM write pointer wraps around the top of the buffer. It remains set until this bit is written 0 in Disabled state. In Software FIFO mode, this flag indicates that the current space in the trace memory is less than or equal to the value programmed in the BUFWM Register, that is, Fill level >= MEM_SIZE - BUFWM. The FULL output from the TMC reflects the value of this register bit, except when the Integration Mode bit in the ITCTRL Register, 0xF00, is set.

9.20.3 css600_tmc_etr RAM Read Data register, RRD

Reading this register allows data to be read from the trace memory at the location pointed to by the RRP/RRPHI registers when either in the Disabled state or operating in Circular Buffer mode (CB) or Software FIFO mode 1 (SWF1).

When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the AXI data width is the same and a memory word holds 4, 8, or 16 bytes. When ATB_DATA_WIDTH is 64 or 128 bit wide multiple RRD reads must be performed to read a full memory word. When a full memory width of data has been read via the RRD register, the RRP register is incremented to the next memory word. When the TMC left the Disabled state and the trace memory is empty, this register returns 0xFFFFFFFF. When the TMC has left the Disabled state and the trace memory is empty, this register returns 0xFFFFFFFF. When operating in Circular Buffer (CB) mode and the TMC left the Disabled state, this register returns 0xFFFFFFFF in all other states except the Stopped state. When operating in Software FIFO mode 2 (SWF2) mode, this register also returns 0xFFFFFFFF. When the MemErr bit in the STS Register is set, reading this register returns an error response on the APB completer interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x010`

Type

RO

Reset value

`0x--------`

Bit descriptions

The following figure shows the RRD register bit assignments.

Figure 9-546: Bit assignment diagram for the RRD register

The following table shows the RRD register bit descriptions.

Table 9-565: RRD bit descriptions

Bits	Name	Reset	Type	Description
31:0	RRD	UNKNOWN	RO	Returns the data read from trace memory

9.20.4 css600_tmc_etr RAM Read Pointer register, RRP

The RRP register contains the value of the read pointer that is used to read entries from trace memory over the APB interface. Software must program it before enabling trace capture.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x014`

Type

RW

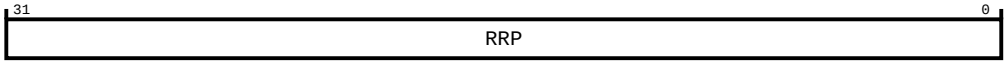
Reset value

`0x--------`

Bit descriptions

The following figure shows the RRP register bit assignments.

Figure 9-547: Bit assignment diagram for the RRP register



The following table shows the RRP register bit descriptions.

Table 9-566: RRP bit descriptions

Bits	Name	Reset	Type	Description
31:0	RRP	**UNKNOWN**	RW	This register represents the lower 32 bits of the AXI address that is used to access the trace memory with RRD read accesses. When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the AXI data width is the same and a memory word holds 4, 8, or 16 bytes. When a full memory width of data has been read via the RRD register, the RRP register is incremented to the next memory word. The lowest 2 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 32 bits. The lowest 3 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 64 bits. The lowest 4 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 128 bits.

9.20.5 css600_tmc_etr RAM Write Pointer register, RWP

The RWP register sets the write pointer that writes entries into the trace memory. Software must program it before enabling trace capture.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x018`

Type

RW

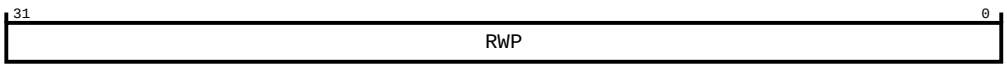
Reset value

`0x--------`

Bit descriptions

The following figure shows the RWP register bit assignments.

Figure 9-548: Bit assignment diagram for the RWP register



The following table shows the RWP register bit descriptions.

Table 9-567: RWP bit descriptions

Bits	Name	Reset	Type	Description
31:0	RWP	UNKNOWN	RW	This register represents the lower 32 bits of the AXI address that is used to access the trace memory with RRD read accesses. When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the AXI data width is the same and a memory word holds 4, 8, or 16 bytes. When a full memory width of data has been written via the RWD register, the RWP register is incremented to the next memory word. The lowest 2 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 32 bits. The lowest 3 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 64 bits. The lowest 4 bits have access type **RAZ/WI** when ATB_DATA_WIDTH is 128 bits. When leaving the Disabled state and RWP does not point inside the buffer space (starting with DBA and ending before DBA+RSZ*4) then STS.MemErr is set.

9.20.6 css600_tmc_etr Trigger Counter register, TRG

The TRG register, in Circular Buffer mode, specifies the number of 32-bit words to capture in the trace memory, after detecting either a rising edge on the trigin input or a trigger packet in the incoming trace stream, that is, where atid_rx = 0x7D.

The value programmed must be aligned to the frame length of 128 bits. Software must program this register before leaving Disabled state.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x01C`

Type

RW

Reset value

`0x--------`

Bit descriptions

The following figure shows the TRG register bit assignments.

Figure 9-549: Bit assignment diagram for the TRG register

The following table shows the TRG register bit descriptions.

Table 9-568: TRG bit descriptions

Bits	Name	Reset	Type	Description
31:0	TRG	UNKNOWN	RW	Trigger count. This count represents the number of 32-bit words of trace that are captured between a trigger packet and a trigger event. The lowest two bits have access type **RAZ/WI**.

9.20.7 css600_tmc_etr Control Register, CTL

The CTL register controls trace stream capture. Setting the CTL.TraceCaptEn bit to 1 enables the TMC to capture the trace data. When trace capture is enabled, the formatter behavior is controlled by the FFCR register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x020`

Type

RW

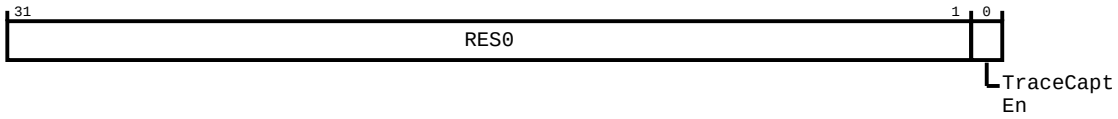
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CTL register bit assignments.

Figure 9-550: Bit assignment diagram for the CTL register



The following table shows the CTL register bit descriptions.

Table 9-569: CTL bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
0	TraceCaptEn	`0b0`	RW	Trace capture enable: **`0b0`** Disable trace capture **`0b1`** Enable trace capture

9.20.8 css600_tmc_etr RAM Write Data register, RWD

The RWD register enables testing of trace memory connectivity to the TMC. Writing this register allows data to be written to the trace memory at the location pointed to by the RWP/RWPHI registers when in the Disabled state.

When ATB_DATA_WIDTH is 32, 64 or 128 bit wide the AXI data width is the same and a memory word holds 4, 8, or 16 bytes. When ATB_DATA_WIDTH is 64 or 128 bit wide multiple RWD writes must be performed to write a full memory word. When a full memory width of data has been written via the RWD register, the data is written to the trace memory and the RWP register is incremented to the next memory word. When the STS.MemErr bit is set, writing to this register returns an error response on the APB interface and the write data is discarded.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x024`

Type

WO

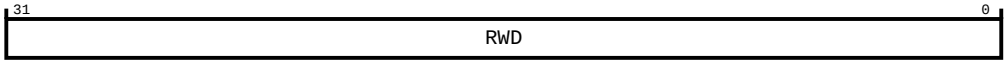
Reset value

`0x00000000`

Bit descriptions

The following figure shows the RWD register bit assignments.

Figure 9-551: Bit assignment diagram for the RWD register



The following table shows the RWD register bit descriptions.

Table 9-570: RWD bit descriptions

Bits	Name	Reset	Type	Description
31:0	RWD	`0x0`	WO	Data written to this register is placed in the trace memory.

9.20.9 css600_tmc_etr Mode register, MODE

The MODE register controls the TMC operating mode. The operating mode can only be changed when the TMC is in Disabled state. Attempting to write to this register in any other state results in **UNPREDICTABLE** behavior. The operating mode is ignored when in Disabled state.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x028`

Type

RW

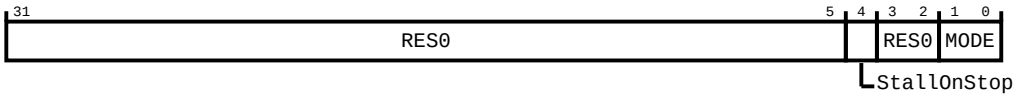
Reset value

`0x000000--`

Bit descriptions

The following figure shows the MODE register bit assignments.

Figure 9-552: Bit assignment diagram for the MODE register



The following table shows the MODE register bit descriptions.

Table 9-571: MODE bit descriptions

Bits	Name	Reset	Type	Description
31:5	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
4	StallOnStop	**UNKNOWN**	RW	Stall On Stop. If this bit is set and the formatter stops as a result of a stop event, the output atready_rx is de-asserted to stall the ATB interface and avoid loss of trace. If this bit is clear and the formatter stops as a result of a stop event, signal atready_rx remains asserted but the TMC discards further incoming trace.
3:2	**RES0**	`0b00`	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
1:0	MODE	UNKNOWN	RW	Selects the operating mode. If a reserved MODE value is programmed and trace capture is enabled, the TMC starts to operate in Software FIFO mode 1 (SWF1). However, reading the MODE.MODE field returns the programmed value. **0b00** CB, Circular Buffer mode. **0b01** SWF1, Software Read FIFO mode 1. **0b10** Reserved. (SWF1) **0b11** SWF2, Software Read FIFO mode 2.

9.20.10 css600_tmc_etr Latched Buffer Fill Level, LBUFLEVEL

Reading the LBUFLEVEL register returns the maximum fill level of the trace memory in 32-bit words since this register was last read. Reading this register also results in its contents being updated to the current fill level. When entering Disabled state, it retains its last value.

While in Disabled state, reads from this register do not affect its value. When exiting Disabled state, the LBUFLEVEL register is updated to the current fill level.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x02C

Type

RO

Reset value

0x--------

Bit descriptions

The following figure shows the LBUFLEVEL register bit assignments.

Figure 9-553: Bit assignment diagram for the LBUFLEVEL register

The following table shows the LBUFLEVEL register bit descriptions.

Table 9-572: LBUFLEVEL bit descriptions

Bits	Name	Reset	Type	Description
31	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
30:0	LBUFLEVEL	**UNKNOWN**	RO	Latched Buffer Fill Level. Indicates the maximum fill level of the trace memory in 32-bit words since this register was last read.

9.20.11 css600_tmc_etr Current Buffer Fill Level, CBUFLEVEL

The CBUFLEVEL register indicates the current fill level of the trace memory in units of 32-bit words. When the TMC leaves Disabled state, this register dynamically indicates the current fill level of trace memory. It retains its value on entering Disabled state.

It is not affected by the reprogramming of pointer registers in Disabled state with the exception of RRD reads and RWD writes.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x030

Type

RO

Reset value

0x--------

Bit descriptions

The following figure shows the CBUFLEVEL register bit assignments.

Figure 9-554: Bit assignment diagram for the CBUFLEVEL register

The following table shows the CBUFLEVEL register bit descriptions.

Table 9-573: CBUFLEVEL bit descriptions

Bits	Name	Reset	Type	Description
31	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
30:0	CBUFLEVEL	**UNKNOWN**	RO	Current Buffer Fill Level. Indicates the current fill level of the trace memory in 32-bit words.

9.20.12 css600_tmc_etr Buffer Level Water Mark, BUFWM

The value that is programmed into the BUFWM register indicates the required threshold vacancy level in 32-bit words in the trace memory.

When the available space in the FIFO is less than or equal to this value, that is, fill level >= (MEM_SIZE - BUFWM), the full output is asserted and the STS.Full bit is set. This register is used only in the FIFO modes, that is, Software FIFO mode 1 (SWF1), Software FIFO mode 2 (SWF2), and Hardware FIFO (HWF) modes. In Circular Buffer (CB) mode, the same functionality is obtained by programming the RWP to the required vacancy trigger level, so that when the pointer wraps around, the full output gets asserted indicating that the vacancy level has fallen below the required level. Reading this register returns the programmed value. The maximum value that can be written into this register is MEM_SIZE - 1, in which case the full output is asserted after the first 32-bit word is written to trace memory. Writing to this register other than when in Disabled state results in **UNPREDICTABLE** behavior. Any software using it must program it with an initial value before setting the CTL.TraceCaptEn bit to 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x034`

Type

RW

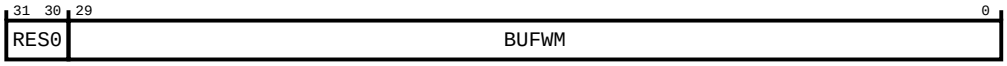
Reset value

`0x--------`

Bit descriptions

The following figure shows the BUFWM register bit assignments.

Figure 9-555: Bit assignment diagram for the BUFWM register



The following table shows the BUFWM register bit descriptions.

Table 9-574: BUFWM bit descriptions

Bits	Name	Reset	Type	Description
31:30	**RES0**	`0b00`	RO	Reserved bit or field with SBZP behavior.
29:0	BUFWM	**UNKNOWN**	RW	Buffer Level Watermark. Indicates the required threshold vacancy level in 32-bit words in the trace memory.

9.20.13 css600_tmc_etr RAM Read Pointer High register, RRPHI

The RAM Read Pointer High register contains address bits >= bit[32] of the read pointer that is used to read entries from trace memory over the APB interface. Software must program it before enabling trace capture.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x038

Type

RW

Reset value

0x--------

Bit descriptions

The following figure shows the RRPHI register bit assignments.

Figure 9-556: Bit assignment diagram for the RRPHI register

The following table shows the RRPHI register bit descriptions.

Table 9-575: RRPHI bit descriptions

Bits	Name	Reset	Type	Description
31:0	RRPHI	UNKNOWN	RW	RAM Read Pointer High. Bits [32] and above of the RAM read pointer.

9.20.14 css600_tmc_etr RAM Write Pointer High register, RWPHI

The RAM Write Pointer High register sets bits >= bit[32] of the write pointer that is used to write entries into the trace memory. Software must program it before enabling trace capture.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x03C

Type

RW

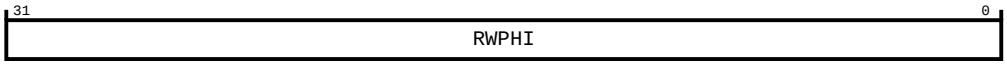
Reset value

0x--------

Bit descriptions

The following figure shows the RWPHI register bit assignments.

Figure 9-557: Bit assignment diagram for the RWPHI register



The following table shows the RWPHI register bit descriptions.

Table 9-576: RWPHI bit descriptions

Bits	Name	Reset	Type	Description
31:0	RWPHI	UNKNOWN	RW	RAM Write Pointer High. Bits [32] and above of the RAM write pointer.

9.20.15 css600_tmc_etr AXI Control Register, AXICTL

This register controls TMC accesses to system memory through the AXI interface. The TMC only performs data accesses, so the arprot_m[2] and awprot_m[2] outputs are LOW for all AXI transfers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x110

Type

RW

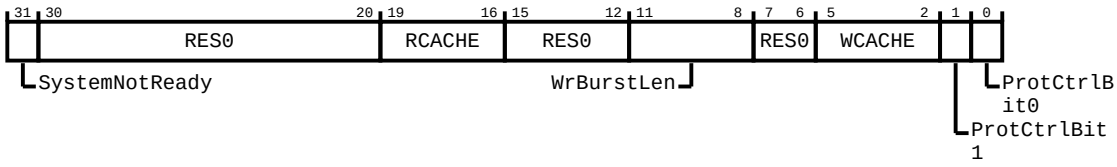
Reset value

0x00000-0-

Bit descriptions

The following figure shows the AXICTL register bit assignments.

Figure 9-558: Bit assignment diagram for the AXICTL register



The following table shows the AXICTL register bit descriptions.

Table 9-577: AXICTL bit descriptions

Bits	Name	Reset	Type	Description
31	SystemNotReady	0b0	RO	System is not ready. **0b0** System is ready. **0b1** System is not ready. ETR is waiting for MMU to respond to first translation stash request. This can only happen when translation stashing is enabled.
30:20	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
19:16	RCACHE	0b0000	RW	This field controls the AXI cache encoding for read transfers, that is, the value to be driven on the arcache_m[3:0] bus. Software must only program a valid AXI cache encoding value in this field. These values are defined in the AMBA AXI Protocol Specification. If software attempts to program an invalid value, 0x0 is written to this field instead.
15:12	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
11:8	WrBurstLen	**UNKNOWN**	RW	Write Burst Length. This field indicates the maximum number of data transfers that can occur within each burst that is initiated by the TMC on the AXI interface. The write burst that is initiated on the AXI can be shorter than the programmed value in a case when the formatter has stopped due to a stop condition having occurred.
7:6	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
5:2	WCACHE	0b0000	RW	This field controls the AXI cache encoding for write transfers, that is, the value to be driven on the awcache_m[3:0] bus. Software must only program a valid AXI cache encoding value in this field. These values are defined in the AMBA AXI Protocol Specification. If software attempts to program an invalid value, 0x0 is written to this field instead.
1	ProtCtrlBit1	**UNKNOWN**	RW	Secure Access (AXI definition). This bit controls the value that is driven on arprot_m[1] or awprot_m[1] on the AXI manager interface when performing AXI transfers.
0	ProtCtrlBit0	**UNKNOWN**	RW	Privileged Access (AXI definition). This bit controls the value that is driven on arprot_m[0] or awprot_m[0] on the AXI manager interface when performing AXI transfers.

9.20.16 css600_tmc_etr Data Buffer Address Low register, DBALO

This register, together with the DBAHI register, enables the TMC to locate the trace data buffer in system memory. This register contains bits[31:0] of the start address of the trace data buffer in system memory. This register is 32 bits wide if AXI_ADDR_WIDTH is >= 32 bits.

Software must program it before enabling trace capture, and the programmed value must be aligned to the Trace Memory Data Width and the Frame Width. Programming an unaligned value

results in **UNPREDICTABLE** behavior. Modifying this register other than when in Disabled state results in **UNPREDICTABLE** behavior.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x118`

Type

RW

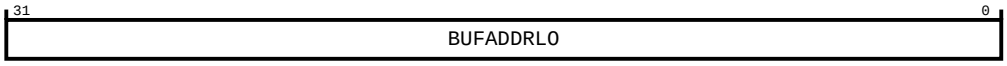
Reset value

`0x--------`

Bit descriptions

The following figure shows the DBALO register bit assignments.

Figure 9-559: Bit assignment diagram for the DBALO register



The following table shows the DBALO register bit descriptions.

Table 9-578: DBALO bit descriptions

Bits	Name	Reset	Type	Description
31:0	BUFADDRLO	**UNKNOWN**	RW	Data Buffer Low Address. Holds the lower 32 bits of the AXI address that is used to locate the trace buffer in system memory. The lowest four bits have access type **RAZ/WI**.

9.20.17 css600_tmc_etr Data Buffer Address HIGH register, DBAHI

This register, together with the DBALO register, enables the TMC to locate the trace data buffer in system memory. It contains bits >= bit[32] of the start address of the trace buffer in system memory.

The width of this register is given by: (AXI_ADDR_WIDTH - 32), however, if AXI_ADDR_WIDTH is <= 32 bits, this register is reserved and access type is **RAZ/WI**. Modifying this register, other than when in Disabled state, results in **UNPREDICTABLE** behavior. Software must program it with an initial value before setting CTL.TraceCaptEn bit to 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x11C

Type

RW

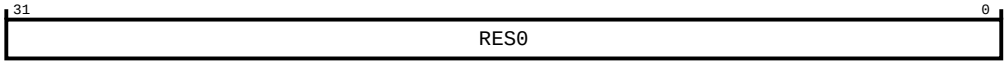
Reset value

0x00000000

Bit descriptions

The following figure shows the DBAHI register bit assignments.

Figure 9-560: Bit assignment diagram for the DBAHI register



The following table shows the DBAHI register bit descriptions.

Table 9-579: DBAHI bit descriptions

Bits	Name	Reset	Type	Description
31:0	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

9.20.18 css600_tmc_etr RAM Update Read Pointer register, RURP

The RURP register enables software to inform the TMC of the amount of trace data that is extracted directly from system memory in Software FIFO mode 2 (SWF2). Writes to this register are ignored when the TMC is in Disabled state or when not in SWF2 mode.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x120

Type

WO

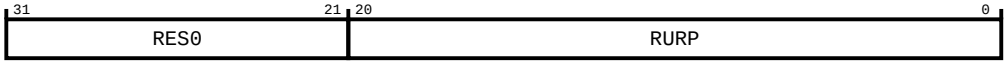
Reset value

0x00000000

Bit descriptions

The following figure shows the RURP register bit assignments.

Figure 9-561: Bit assignment diagram for the RURP register



The following table shows the RURP register bit descriptions.

Table 9-580: RURP bit descriptions

Bits	Name	Reset	Type	Description
31:21	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
20:0	RURP	0x0	WO	RAM Update Read Pointer. A write to the RURP register causes the TMC to update the RAM Read Pointer, both the RRP and RRPHI registers, based on the value that is written to it. RURP allows up to 1MB of data to be extracted in a single chunk. Reads always return 0x0. The following constraints apply to the write values: 0x000000 - no effect, 0x000010-0x100000 - increment RRP by this value, 0x100010-0x1FFFFF - reserved. The programmed value must also be aligned to the Trace Memory Data Width and the Frame Width. Programming an unaligned or reserved value results in **UNPREDICTABLE** behavior.

9.20.19 css600_tmc_etr Formatter and Flush Status Register, FFSR

This register indicates the status of the Formatter, and the status of Flush request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x300

Type

RO

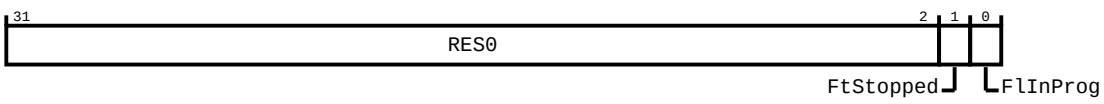
Reset value

0x0000000-

Bit descriptions

The following figure shows the FFSR register bit assignments.

Figure 9-562: Bit assignment diagram for the FFSR register



The following table shows the FFSR register bit descriptions.

Table 9-581: FFSR bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	FtStopped	**UNKNOWN**	RO	Formatter Stopped. This bit behaves the same way as STS.FtEmpty.
0	FlInProg	**UNKNOWN**	RO	Flush In Progress. This bit indicates whether the TMC is currently processing a flush request. The flush initiation is controlled by the flush control bits in the FFCR register. This bit is cleared to 0 when leaving the Disabled state and retains its value when entering the Disabled state. When in Disabled state, this bit is not updated. **0b0** No flush activity in progress. **0b1** Flush in progress on the ATB receiver interface or the TMC internal pipeline.

9.20.20 css600_tmc_etr Formatter and Flush Control Register, FFCR

The FFCR controls the generation of stop, trigger and flush events. The insertion of a flush completion packet and the insertion of a trigger packet in the formatted trace is enabled here.

Also one of the 2 formatter modes for bypass mode and normal mode can be changed here when the formatter has stopped.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x304

Type

RW

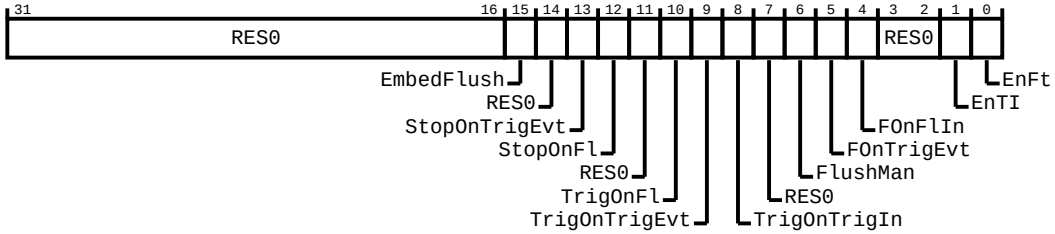
Reset value

0x00000000

Bit descriptions

The following figure shows the FFCR register bit assignments.

Figure 9-563: Bit assignment diagram for the FFCR register



The following table shows the FFCR register bit descriptions.

Table 9-582: FFCR bit descriptions

Bits	Name	Reset	Type	Description
31:16	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
15	EmbedFlush	0b0	RW	Embed Flush ID (flush completion packet). Enables insertion of Flush ID 0x7B with a single byte of data payload = 0x00 in the output trace, immediately after the last flush data byte, when a flush completes on the ATB receiver interface. This bit is effective only in Normal formatting modes. In Bypass mode, the Flush ID insertion remains disabled and this bit is ignored. **0b0** Disable Flush ID insertion. **0b1** Enable Flush ID insertion.
14	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
13	StopOnTrigEvt	0b0	RW	Stop On Trigger Event. If this bit is set, the formatter is stopped when a Trigger Event has been observed. This bit must be used only in Circular Buffer (CB) mode. If trace capture is enabled in Software FIFO mode 1 (SWF1), or Software FIFO mode 2 (SWF2) with this bit set, it results in **UNPREDICTABLE** behavior.
12	StopOnFl	0b0	RW	Stop On Flush. If this bit is set, the formatter is stopped on completion of a flush operation. The initiation of a flush operation is controlled by programming the register bits FFCR.FlushMan, FFCR.FOnTrigEvt, and FFCR.FOnFlIn. When a flush-initiation condition occurs, afvalid_rx is asserted, and when the flush completion is received, that is, afready_rx=1, trace capture is stopped. Any remaining data in the formatter is appended with a post-amble and written to trace memory. The flush operation is then complete.
11	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
10	TrigOnFl	0b0	RW	Indicate on trace stream the completion of flush. If this bit is set, a trigger is indicated on the trace stream when afready_rx is received for a flush in progress. If this bit is clear, no triggers are embedded in the trace stream on flush completion. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit.
9	TrigOnTrigEvt	0b0	RW	Indicate on trace stream the occurrence of a Trigger Event. If this bit is set, a trigger is indicated on the output trace stream when a Trigger Event occurs. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit. This bit must be used only in CB mode. If trace capture is enabled in SWF1, or SWF2 mode with this bit set, it results in **UNPREDICTABLE** behavior.
8	TrigOnTrigIn	0b0	RW	Indicate on trace stream the occurrence of a rising edge on trigin. If this bit is set, a trigger is indicated on the trace stream when a rising edge is detected on the trigin input. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit.
7	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
6	FlushMan	0b0	RW	Manually generate a flush of the system. Writing 1 to this bit causes a flush to be generated. This bit is cleared automatically when, in formatter bypass mode, afready_rx was sampled high, or, in normal formatting mode, afready_rx was sampled high and all flush data was output to the trace memory. If CTL.TraceCaptEn=0, writes to this bit are ignored.
5	FOnTrigEvt	0b0	RW	Flush on Trigger Event. If FFCR.StopOnTrigEvt is set, this bit is ignored. Setting this bit generates a flush when a Trigger Event occurs. If FFCR.StopOnTrigEvt is set, this bit is ignored. This bit must be used only in CB mode. If trace capture is enabled in SWF1, or SWF2 mode with this bit set, it results in **UNPREDICTABLE** behavior.
4	FOnFlIn	0b0	RW	Setting this bit enables the detection of transitions on the flushin input by the TMC. If this bit is set and the formatter has not already stopped, a rising edge on flushin initiates a flush request.
3:2	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
1	EnTI	0b0	RW	Enable Trigger Insertion. Setting this bit enables the insertion of triggers in the formatted trace stream. A trigger is indicated by inserting one byte of data 0x00 with atid_rx=0x7D in the trace stream. Trigger indication on the trace stream is also controlled by the register bits FFCR.TrigOnFl, FFCR.TrigOnTrigEvt, and FFCR.TrigOnTrigIn. This bit can only be changed when the TMC is in Disabled state. If FFCR.EnTI bit is set formatting is enabled.
0	EnFt	0b0	RW	Enable Formatter. If this bit is set, formatting is enabled. When EnTi is set, formatting is enabled. When CB mode is not used, formatting is also enabled. For backwards-compatibility with earlier versions of the ETB disabling of formatting is supported only in CB mode. This bit can only be changed when TMC is in Disabled state.

9.20.21 css600_tmc_etr Periodic Synchronization Counter Register, PSCR

This register determines the reload value of the Periodic Synchronization Counter. This counter enables the frequency of sync packets to be optimized to the trace capture buffer size. The default behavior of the counter is to generate periodic synchronization requests, syncreq_rx, on the ATB receiver interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x308

Type

RW

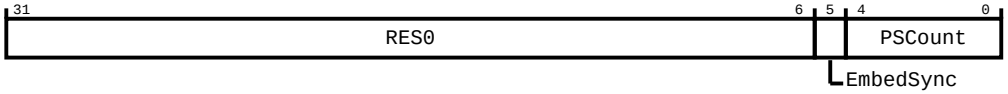
Reset value

0x0000000A

Bit descriptions

The following figure shows the PSCR register bit assignments.

Figure 9-564: Bit assignment diagram for the PSCR register



The following table shows the PSCR register bit descriptions.

Table 9-583: PSCR bit descriptions

Bits	Name	Reset	Type	Description
31:6	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
5	EmbedSync	`0b0`	RW	Embed Frame Sync Packet in the trace stream. Setting this bit to 1 enables the formatter to insert frame sync packets in the trace stream at periodic intervals. If this bit is set and the Synchronization Counter is enabled, the formatter inserts a 32-bit frame sync packet in the trace stream when the counter reaches 0. This bit is effective only when formatting is enabled, that is when FFCR.EnTI=1 or FFCR.EnFt=1, and it is ignored when the formatter is in bypass mode.
4:0	PSCount	`0b01010`	RW	Periodic Synchronization Count. Determines the reload value of the Synchronization Counter. The reload value takes effect the next time the counter reaches zero. When trace capture is enabled, the Synchronization Counter counts the number of bytes of trace data that is stored into the trace memory, regardless of whether the trace data has been formatted by the TMC or not, since the occurrence of the last sync request on the ATB receiver interface. When the counter reaches 0, a sync request is sent on the ATB receiver interface. Reads from this register return the reload value that is programmed in this register. This field resets to `0x0A`, that is, the default sync period is 2^10 bytes. If a reserved value is programmed in this register field, the value `0x1B` is used instead, and subsequent reads from this register also return `0x1B`. The following constraints apply to the values written to the PSCount field: `0x0 - synchronization is disabled, 0x1-0x6` - reserved, `0x7-0x1B` - synchronization period is 2^PSCount bytes. The smallest value `0x7` gives a sync period of 128 bytes. The maximum allowed value `0x1B` gives a sync period of 2^27 bytes, `0x1C-0x1F` - reserved.

9.20.22 css600_tmc_etr AXI Control Register 1, AXICTL1

This register controls address translation stashing requests made to a system MMU through the AXI interface. This register can be written in Disabled state. In all other states, writing this register is disabled.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xED0`

Type

RW

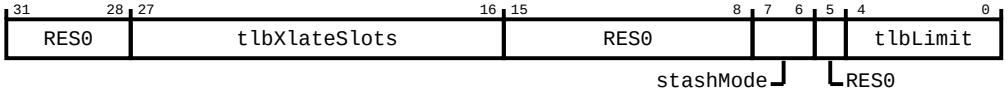
Reset value

`0x0---00--`

Bit descriptions

The following figure shows the AXICTL1 register bit assignments.

Figure 9-565: Bit assignment diagram for the AXICTL1 register



The following table shows the AXICTL1 register bit descriptions.

Table 9-584: AXICTL1 bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:16	tlbXlateSlots	**UNKNOWN**	RW	0x02 - 0xFFF ETR issues a maximum of 2 - 4095 outstanding stash translation requests. Values less than 0x2 are treated as 0x2. The reset default is the stash_xlate_limit tie-off value.
15:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:6	stashMode	**UNKNOWN**	RW	**0x00** Translation stashing is disabled. **0x01** if tlbLimit>0, Translation stashing is enabled in Direct indexing mode. The ETR does not provide prior notice to evict an old TLB entry. Instead the ETR instructs fetching a TLB entry with a stash translation transaction along with an index (provided on awuser_m) to override an old TLB entry. **0x02** if tlbLimit>0, Translation stashing is enabled in Unstashing mode. Before fetching a new TLB entry with an stash translation transaction, the ETR issues the eviction of an old TLB entry with an unstash translation transaction. Values larger than 0x2 are treated as 0x2. The reset default is the stash_mode tie-off value.
5	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
4:0	tlbLimit	**UNKNOWN**	RW	**0x00 |** Translation stashing is disabled. || 0x01 - 0x10 | if stashMode>0, the ETR issues stash translation requests for up to 2^tlbLimit * 4k pages ahead of RWP page. Values larger than 0x10 are treated as 0x10. The reset default is the tlb_limit tie-off value.

9.20.23 css600_tmc_etr Integration Test Event and Interrupt Control Register, ITEVTINTR

This register controls the values of event and interrupt outputs in integration mode. In functional mode, this register behaves as **RAZ/WI**.

In integration mode, the value that is written to any bit of this register is driven on the output pin that is controlled by that bit and the reads return 0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEE0`

Type

WO

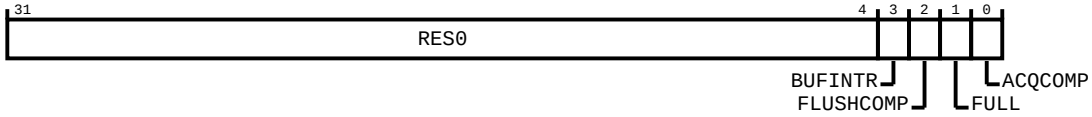
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITEVTINTR register bit assignments.

Figure 9-566: Bit assignment diagram for the ITEVTINTR register



The following table shows the ITEVTINTR register bit descriptions.

Table 9-585: ITEVTINTR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
3	BUFINTR	`0b0`	WO	Controls the value of bufintr output in integration mode.
2	FLUSHCOMP	`0b0`	WO	Controls the value of flushcomp output in integration mode.
1	FULL	`0b0`	WO	Controls the value of full output in integration mode.
0	ACQCOMP	`0b0`	WO	Controls the value of acqcomp output in integration mode.

9.20.24 css600_tmc_etr Integration Test Trigger In and Flush In register, ITTRFLIN

This register captures the values of the flushin and trigin inputs in integration mode. In functional mode, this register behaves as **RAZ/WI**.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEE8

Type

RO

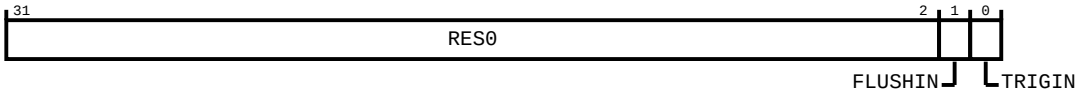
Reset value

0x00000000

Bit descriptions

The following figure shows the ITTRFLIN register bit assignments.

Figure 9-567: Bit assignment diagram for the ITTRFLIN register



The following table shows the ITTRFLIN register bit descriptions.

Table 9-586: ITTRFLIN bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	FLUSHIN	0b0	RO	Integration status of flushin input. In integration mode, this bit latches to 1 on a rising edge of the flushin input. It is cleared when the register is read or when integration mode is disabled.
0	TRIGIN	0b0	RO	Integration status of trigin input. In integration mode, this bit latches to 1 on a rising edge of the trigin input. It is cleared when the register is read or when integration mode is disabled.

9.20.25 css600_tmc_etr Integration Test ATB Data 0 Register, ITATBDATA0

This register captures the value of atdata_rx input in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of corresponding atdata_rx bits. The width of this register is given by: 1+(ATB DATA WIDTH)/8.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEEC

Type

RO

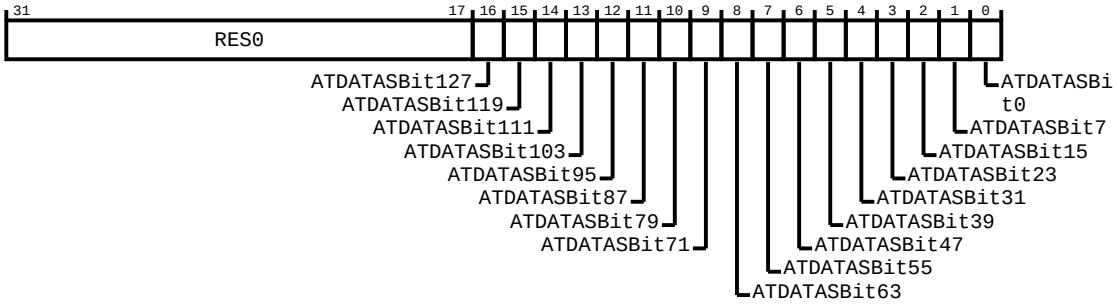
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBDATA0 register bit assignments.

Figure 9-568: Bit assignment diagram for the ITATBDATA0 register



The following table shows the ITATBDATA0 register bit descriptions.

Table 9-587: ITATBDATA0 bit descriptions

Bits	Name	Reset	Type	Description
31:17	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
16	ATDATASBit127	0b0	RO	Returns the value of atdata_rx[127] input in integration mode. RES0 if ATB_DATA_WIDTH < 128.
15	ATDATASBit119	0b0	RO	Returns the value of atdata_rx[119] input in integration mode. RES0 if ATB_DATA_WIDTH < 128.
14	ATDATASBit111	0b0	RO	Returns the value of atdata_rx[111] input in integration mode. RES0 if ATB_DATA_WIDTH < 128.
13	ATDATASBit103	0b0	RO	Returns the value of atdata_rx[103] input in integration mode. RES0 if ATB_DATA_WIDTH < 128.
12	ATDATASBit95	0b0	RO	Returns the value of atdata_rx[95] input in integration mode. RES0 if ATB_DATA_WIDTH < 128.
11	ATDATASBit87	0b0	RO	Returns the value of atdata_rx[87] input in integration mode. RES0 if ATB_DATA_WIDTH < 128.
10	ATDATASBit79	0b0	RO	Returns the value of atdata_rx[79] input in integration mode. RES0 if ATB_DATA_WIDTH < 128.
9	ATDATASBit71	0b0	RO	Returns the value of atdata_rx[71] input in integration mode. RES0 if ATB_DATA_WIDTH < 128.
8	ATDATASBit63	0b0	RO	Returns the value of atdata_rx[63] input in integration mode. RES0 if ATB_DATA_WIDTH < 64.
7	ATDATASBit55	0b0	RO	Returns the value of atdata_rx[55] input in integration mode. RES0 if ATB_DATA_WIDTH < 64.
6	ATDATASBit47	0b0	RO	Returns the value of atdata_rx[47] input in integration mode. RES0 if ATB_DATA_WIDTH < 64.
5	ATDATASBit39	0b0	RO	Returns the value of atdata_rx[39] input in integration mode. RES0 if ATB_DATA_WIDTH < 64.
4	ATDATASBit31	0b0	RO	Returns the value of atdata_rx[31] input in integration mode.
3	ATDATASBit23	0b0	RO	Returns the value of atdata_rx[23] input in integration mode.
2	ATDATASBit15	0b0	RO	Returns the value of atdata_rx[15] input in integration mode.
1	ATDATASBit7	0b0	RO	Returns the value of atdata_rx[7] input in integration mode.
0	ATDATASBit0	0b0	RO	Returns the value of atdata_rx[0] input in integration mode.

9.20.26 css600_tmc_etr Integration Test ATB Control 2 Register, ITATBCTR2

This register enables control of ATB receiver outputs atready_rx, afvalid_rx, and syncreq_rx in integration mode. In functional mode, this register behaves as **RAZ/WI**.

In integration mode, the value that is written to any bit of this register is driven on the output pin that is controlled by that bit and the reads return 0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEF0`

Type

WO

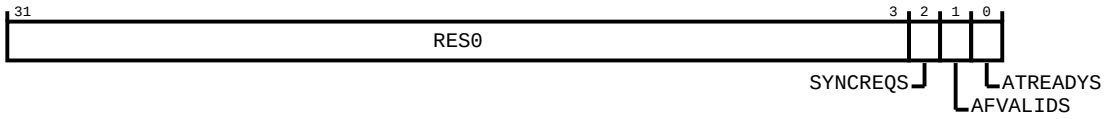
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITATBCTR2 register bit assignments.

Figure 9-569: Bit assignment diagram for the ITATBCTR2 register



The following table shows the ITATBCTR2 register bit descriptions.

Table 9-588: ITATBCTR2 bit descriptions

Bits	Name	Reset	Type	Description
31:3	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
2	SYNCREQS	`0b0`	WO	Controls the value of syncreq_rx output in integration mode.
1	AFVALIDS	`0b0`	WO	Controls the value of afvalid_rx output in integration mode.
0	ATREADYS	`0b0`	WO	Controls the value of atready_rx output in integration mode.

9.20.27 css600_tmc_etr Integration Test ATB Control 1 Register, ITATBCTR1

This register captures the value of the atid_rx[6:0] input in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of atid_rx input.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEF4`

Type

RO

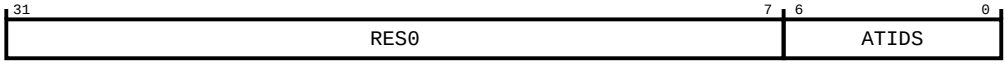
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITATBCTR1 register bit assignments.

Figure 9-570: Bit assignment diagram for the ITATBCTR1 register



The following table shows the ITATBCTR1 register bit descriptions.

Table 9-589: ITATBCTR1 bit descriptions

Bits	Name	Reset	Type	Description
31:7	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
6:0	ATIDS	`0x0`	RO	Returns the value of atid_rx[6:0] input in integration mode.

9.20.28 css600_tmc_etr Integration Test ATB Control 0 Register, ITATBCTR0

This register captures the values of ATB receiver inputs atvalid_rx, afready_rx, atwakeup_rx, and atbytes_rx in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of corresponding input pins.

The width of this register is given by: 8+log2(ATB DATA WIDTH/8).

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEF8`

Type

RO

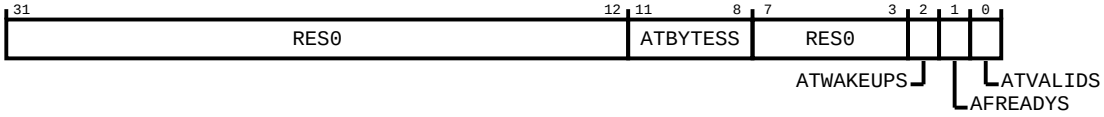
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITATBCTR0 register bit assignments.

Figure 9-571: Bit assignment diagram for the ITATBCTR0 register



The following table shows the ITATBCTR0 register bit descriptions.

Table 9-590: ITATBCTR0 bit descriptions

Bits	Name	Reset	Type	Description
31:12	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
11:8	ATBYTESS	`0b0000`	RO	Returns the value of atbytes_rx input in integration mode. N=8+log2(ATB DATA WIDTH/8).
7:3	**RES0**	`0b00000`	RO	Reserved bit or field with SBZP behavior.
2	ATWAKEUPS	`0b0`	RO	Returns the value of atwakeup_rx input in integration mode.
1	AFREADYS	`0b0`	RO	Returns the value of afready_rx input in integration mode.
0	ATVALIDS	`0b0`	RO	Returns the value of atvalid_rx input in integration mode.

9.20.29 css600_tmc_etr Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

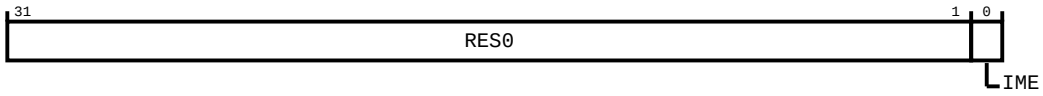
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-572: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-591: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.20.30 css600_tmc_etr Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

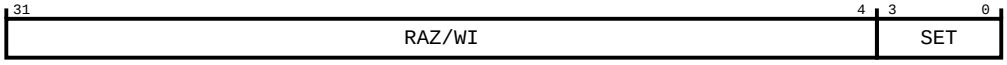
Reset value

0x0000000F

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-573: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-592: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:4	RAZ/WI	0x0	RO	RAZ/WI.
3:0	SET	0b1111	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.20.31 css600_tmc_etr Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA4

Type

RW

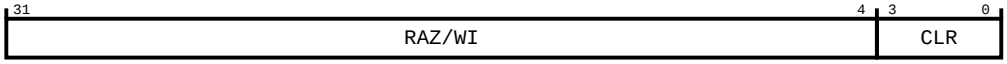
Reset value

0x00000000

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-574: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-593: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	0x0	RO	**RAZ/WI**.
3:0	CLR	0b0000	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.20.32 css600_tmc_etr Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

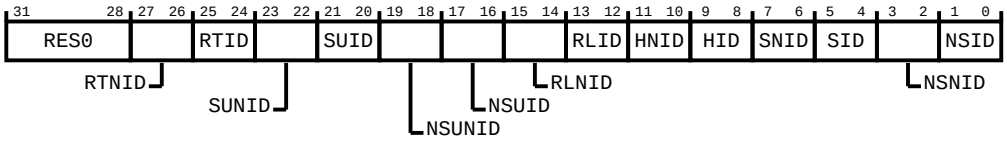
Reset value

0x000000--

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-575: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-594: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	0b00	RO	Root non-invasive debug. **0b00** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.
25:24	RTID	0b00	RO	Root invasive debug. **0b00** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	0b00	RO	Secure non-invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
5:4	SID	**UNKNOWN**	RO	Secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.
3:2	NSNID	0b00	RO	Non-secure non-invasive debug. **0b00** Debug level is not supported.
1:0	NSID	**UNKNOWN**	RO	Non-secure invasive debug. **0b10** Supported and disabled. **0b11** Supported and enabled.

9.20.33 css600_tmc_etr Device Configuration Register 1, DEVID1

Contains an **IMPLEMENTATION DEFINED** value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC4

Type

RO

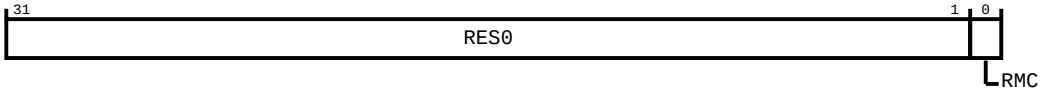
Reset value

0x00000001

Bit descriptions

The following figure shows the DEVID1 register bit assignments.

Figure 9-576: Bit assignment diagram for the DEVID1 register



The following table shows the DEVID1 register bit descriptions.

Table 9-595: DEVID1 bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	RMC	0b1	RO	Register management mode. TMC implements register management mode 1.

9.20.34 css600_tmc_etr Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC8

Type

RO

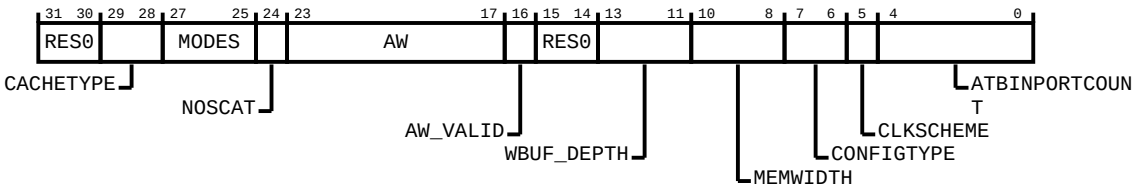
Reset value

0x03----40

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-577: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-596: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:30	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
29:28	CACHETYPE	0b00	RO	Indicates the format of BUSCTL register bus control fields. Reads as 2'b00 indicating that AXICTL bus attribute bits [19:16] and [5:2] follow an implementation-defined non-generic format. See AXICTL register description.

Bits	Name	Reset	Type	Description
27:25	MODES	0b001	RO	Indicates the supported modes of operation. Reads as 3'b001 indicating that ETR supports Circular Buffer (CB), Software FIFO mode 1 (SWF1), and Software FIFO mode 2 (SWF2) modes.
24	NOSCAT	0b1	RO	Indicates whether the scatter-gather mode is implemented. Fixed at 1 indicating that scatter-gather mode is not implemented.
23:17	AW	**IMPLEMENTATION DEFINED**	RO	This field indicates the width of AXI address bus in ETR configuration. This field is valid only when DEVID.AW_VALID is set. Possible values are: **0x20** 32-bit AXI address bus. **0x28** 40-bit AXI address bus. **0x2C** 44-bit AXI address bus. **0x30** 48-bit AXI address bus. **0x34** 52-bit AXI address bus. **0x40** 64-bit AXI address bus.
16	AW_VALID	0b1	RO	Indicates whether field DEVID.AW is valid. The value of this field is fixed at 1.
15:14	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
13:11	WBUF_DEPTH	**IMPLEMENTATION DEFINED**	RO	Log2 of the number of write buffer entries. This value is set by the parameter WBUFFER_DEPTH. Each entry is of size ATB_DATA_WIDTH. **0b010** Depth of Write buffer is 4 entries. **0b011** Depth of Write buffer is 8 entries. **0b100** Depth of Write buffer is 16 entries. **0b101** Depth of Write buffer is 32 entries. **0b110** Depth of Write buffer is 64 entries. **0b111** Depth of Write buffer is 128 entries.
10:8	MEMWIDTH	**IMPLEMENTATION DEFINED**	RO	Indicates the width of the internal memory data bus. For the ETR this value is equal to ATB_DATA_WIDTH. **0b010** Memory interface databus is 32 bits wide. (ATB_DATA_WIDTH = 32bit) **0b011** Memory interface databus is 64 bits wide. (ATB_DATA_WIDTH = 64bit) **0b100** Memory interface databus is 128 bits wide. (ATB_DATA_WIDTH = 128bit)

Bits	Name	Reset	Type	Description
7:6	CONFIGTYPE	0b01	RO	Indicates the TMC configuration. **0b01** ETR - Embedded Trace Router
5	CLKSCHEME	0b0	RO	RAM Clocking Scheme. This value indicates the TMC RAM clocking scheme used, that is, whether the TMC RAM operates synchronously or asynchronously to the TMC clock. Fixed to 0 indicating that TMC RAM clock is synchronous to the clk input.
4:0	ATBINPORTCOUNT	0b00000	RO	Hidden Level of ATB input multiplexing. This value indicates the type/number of ATB multiplexing present on the input ATB. Fixed to 0x00 indicating that no multiplexing is present.

9.20.35 css600_tmc_etr Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

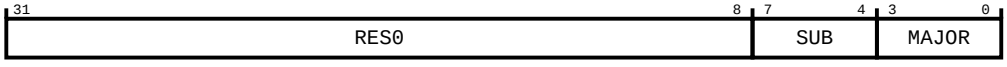
Reset value

0x00000021

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-578: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-597: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	0b0010	RO	Minor classification. Returns 0x2, indicating this component is a Buffer.
3:0	MAJOR	0b0001	RO	Major classification. Returns 0x1, indicating this component is a Trace Sink.

9.20.36 css600_tmc_etr Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD0`

Type

RO

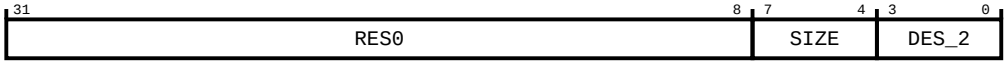
Reset value

`0x00000004`

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-579: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-598: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	`0b0000`	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	`0b0100`	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.20.37 css600_tmc_etr Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

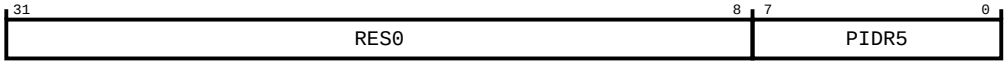
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-580: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-599: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.20.38 css600_tmc_etr Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD8

Type

RO

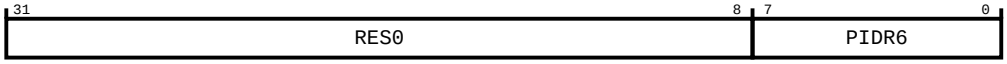
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-581: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-600: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.20.39 css600_tmc_etr Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

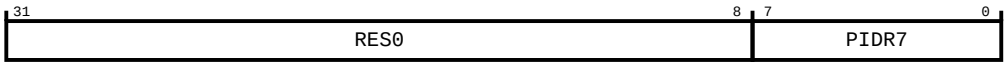
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-582: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-601: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.20.40 css600_tmc_etr Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFE0`

Type

RO

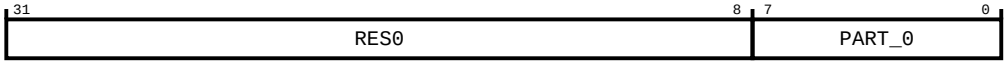
Reset value

`0x000000E8`

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-583: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-602: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	`0xE8`	RO	Part number (lower 8 bits). **`0xE8`** ETR or ETS - Embedded Trace Router or Streamer

9.20.41 css600_tmc_etr Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

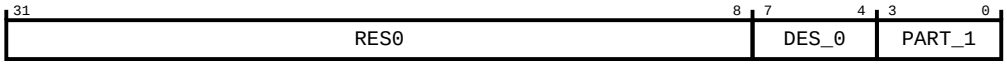
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-584: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-603: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.20.42 css600_tmc_etr Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

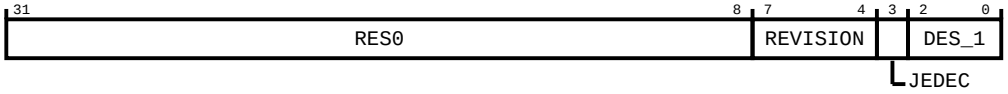
Reset value

0x0000009B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-585: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-604: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	`0b1001`	RO	Revision. It is an incremental value starting at `0x0` for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	`0b1`	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	`0b011`	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.20.43 css600_tmc_etr Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

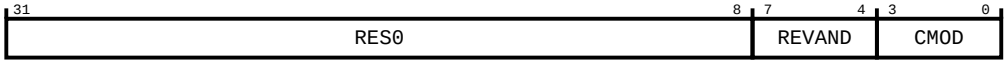
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-586: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-605: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.20.44 css600_tmc_etr Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

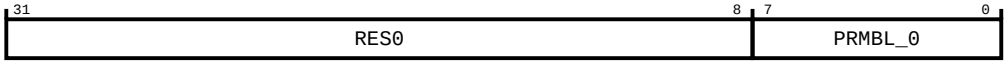
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-587: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-606: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.20.45 css600_tmc_etr Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

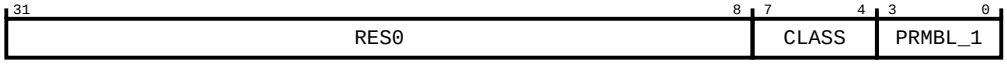
Reset value

0x00000090

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-588: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-607: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1001	RO	Component class **0b1001** CoreSight component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.20.46 css600_tmc_etr Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF8`

Type

RO

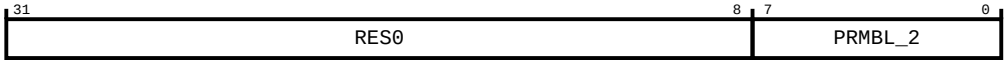
Reset value

`0x00000005`

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-589: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-608: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	`0x5`	RO	Preamble. Returns `0x05`.

9.20.47 css600_tmc_etr Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFFC`

Type

RO

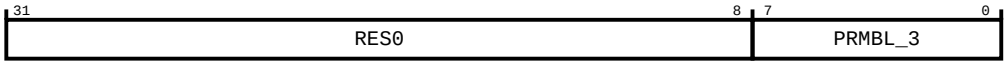
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-590: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-609: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.21 css600_tmc_ets register summary

This section describes the css600_tmc_ets_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-610: css600_tmc_ets_registers register summary

Offset	Name	Type	Reset	Width	Description
0x004	RSZ	RO	0x0000000-	32-bit	RAM Size register
0x00c	STS	RW	0x0000001-	32-bit	Status register
0x010	RRD	RO	0xFFFFFFFF	32-bit	RAM Read Data register
0x01c	TRG	RW	0x--------	32-bit	Trigger Counter register
0x020	CTL	RW	0x00000000	32-bit	Control Register
0x028	MODE	RW	0x000000-0	32-bit	Mode register
0x300	FFSR	RO	0x0000000-	32-bit	Formatter and Flush Status Register
0x304	FFCR	RW	0x00000000	32-bit	Formatter and Flush Control Register
0x308	PSCR	RW	0x0000000A	32-bit	Periodic Synchronization Counter Register
0xee0	ITEVTINTR	WO	0x00000000	32-bit	Integration Test Event and Interrupt Control Register
0xee8	ITTRFLIN	RO	0x00000000	32-bit	Integration Test Trigger In and Flush In register
0xeec	ITATBDATA0	RO	0x00000000	32-bit	Integration Test ATB Data 0 Register
0xef0	ITATBCTR2	WO	0x00000000	32-bit	Integration Test ATB Control 2 Register

Offset	Name	Type	Reset	Width	Description
0xef4	ITATBCTR1	RO	0x00000000	32-bit	Integration Test ATB Control 1 Register
0xef8	ITATBCTR0	RO	0x00000000	32-bit	Integration Test ATB Control 0 Register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x0000000F	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x00000000	32-bit	Authentication Status Register
0xfc4	DEVID1	RO	0x00000001	32-bit	Device Configuration Register 1
0xfc8	DEVID	RO	0x04010-C0	32-bit	Device Configuration Register
0xfcc	DEVTYPE	RO	0x00000021	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000E8	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000009B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.21.1 css600_tmc_ets RAM Size register, RSZ

The RSZ register defines the size of trace memory in units of 32-bit words.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x004

Type

RO

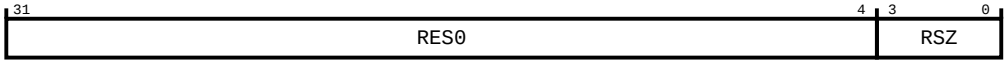
Reset value

0x0000000-

Bit descriptions

The following figure shows the RSZ register bit assignments.

Figure 9-591: Bit assignment diagram for the RSZ register



The following table shows the RSZ register bit descriptions.

Table 9-611: RSZ bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
3:0	RSZ	**IMPLEMENTATION DEFINED**	RO	RAM Size. Indicates the size of trace memory in 32-bit words. Trace memory size is fixed as one AXI Stream data word = log2(ATB_DATA_WIDTH/8).

9.21.2 css600_tmc_ets Status register, STS

The STS register indicates the status of the Trace Memory Controller. After a reset, software must ignore all the fields of this register except STS.TMCReady. The other fields only have meaning when the TMC has left the Disabled state. Writes to all RO fields of this register are ignored.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x00C

Type

RW

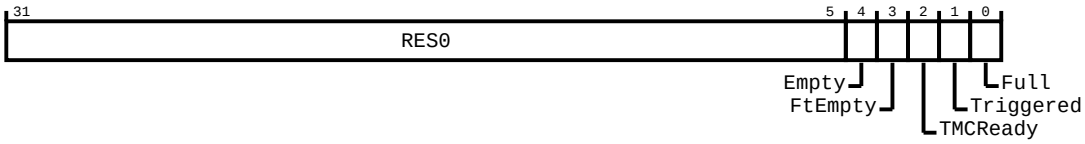
Reset value

0x0000001-

Bit descriptions

The following figure shows the STS register bit assignments.

Figure 9-592: Bit assignment diagram for the STS register



The following table shows the STS register bit descriptions.

Table 9-612: STS bit descriptions

Bits	Name	Reset	Type	Description
31:5	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
4	Empty	0b1	RO	Trace buffer empty. If set, this bit indicates that the trace memory does not contain any valid trace data. The reset value of this bit is 1. On leaving Disabled state, this bit dynamically indicates the empty status of trace memory, !STS_FULL. However, this does not mean that the pipeline stages within the TMC are empty. To determine whether the internal pipeline stages are empty, the software must read the STS.TMCReady bit.
3	FtEmpty	0b1	RO	Trace capture has been completed and all captured trace data has been written to the AXI stream interface, set in Stopped or Disabled state. Otherwise, it is cleared. The reset value is 1
2	TMCReady	0b0	RO	Trace capture has been completed and all captured trace data has been written to the AXI stream interface
1	Triggered	**UNKNOWN**	RO	TMC triggered. This bit is set when trace capture is in progress and the TMC has detected a trigger event. This bit is cleared to 0 when leaving the Disabled state and retains its value when entering the Disabled state. A trigger event is when the TMC has written a set number of data words, as programmed in the TRG register, into the trace memory after a rising edge of trigin input, or a trigger packet (atid_rx = 0x7D) is received in the input trace.
0	Full	**UNKNOWN**	RW	Trace memory full. Because there are no memory pointers to manage, this bit indicates that at least one data transfer has taken place on the AXI Stream interface. It is set when trace capture has stopped. Writes to this bit are allowed in Disabled state, for example for clearing this bit. The FULL output from the TMC reflects the value of this register bit, except when the Integration Mode bit in the ITCTRL Register, 0xF00, is set.

9.21.3 css600_tmc_ets RAM Read Data register, RRD

The RRD register always returns 0xFFFFFFFF.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x010

Type

RO

Reset value

0xFFFFFFFF

Bit descriptions

The following figure shows the RRD register bit assignments.

Figure 9-593: Bit assignment diagram for the RRD register

The following table shows the RRD register bit descriptions.

Table 9-613: RRD bit descriptions

Bits	Name	Reset	Type	Description
31:0	RRD	0xFFFFFFFF	RO	Returns the data read from trace memory

9.21.4 css600_tmc_ets Trigger Counter register, TRG

The TRG register, in Circular Buffer mode, specifies the number of 32-bit words to capture in the trace memory, after detecting either a rising edge on the trigin input or a trigger packet in the incoming trace stream, that is, where atid_rx = 0x7D.

The value programmed must be aligned to the frame length of 128 bits. Software must program this register before leaving Disabled state.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x01C

Type

RW

Reset value

0x--------

Bit descriptions

The following figure shows the TRG register bit assignments.

Figure 9-594: Bit assignment diagram for the TRG register

The following table shows the TRG register bit descriptions.

Table 9-614: TRG bit descriptions

Bits	Name	Reset	Type	Description
31:0	TRG	UNKNOWN	RW	Trigger count. This count represents the number of 32-bit words of trace that are captured between a trigger packet and a trigger event. The lowest two bits have access type **RAZ/WI**.

9.21.5 css600_tmc_ets Control Register, CTL

The CTL register controls trace stream capture. Setting the CTL.TraceCaptEn bit to 1 enables the TMC to capture the trace data. When trace capture is enabled, the formatter behavior is controlled by the FFCR register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x020`

Type

RW

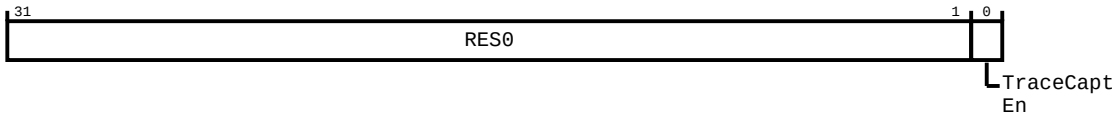
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CTL register bit assignments.

Figure 9-595: Bit assignment diagram for the CTL register



The following table shows the CTL register bit descriptions.

Table 9-615: CTL bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	TraceCaptEn	0b0	RW	Trace capture enable: **0b0** Disable trace capture **0b1** Enable trace capture

9.21.6 css600_tmc_ets Mode register, MODE

The MODE register controls the TMC operating mode. The operating mode can only be changed when the TMC is in Disabled state. Attempting to write to this register in any other state results in **UNPREDICTABLE** behavior. The operating mode is ignored when in Disabled state.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x028`

Type

RW

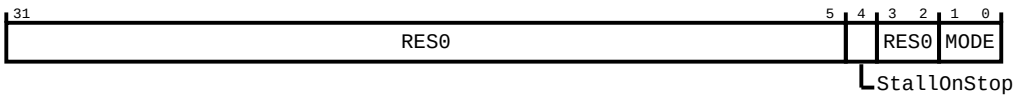
Reset value

`0x000000-0`

Bit descriptions

The following figure shows the MODE register bit assignments.

Figure 9-596: Bit assignment diagram for the MODE register



The following table shows the MODE register bit descriptions.

Table 9-616: MODE bit descriptions

Bits	Name	Reset	Type	Description
31:5	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
4	StallOnStop	**UNKNOWN**	RW	Stall On Stop. If this bit is set and the formatter stops as a result of a stop event, the output atready_rx is de-asserted to stall the ATB interface and avoid loss of trace. If this bit is clear and the formatter stops as a result of a stop event, signal atready_rx remains asserted but the TMC discards further incoming trace.
3:2	**RES0**	`0b00`	RO	Reserved bit or field with SBZP behavior.
1:0	MODE	`0b00`	RO	Fixed to 0 since TMC always operates in Circular Buffer mode in this configuration.

9.21.7 css600_tmc_ets Formatter and Flush Status Register, FFSR

This register indicates the status of the Formatter, and the status of Flush request.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x300

Type

RO

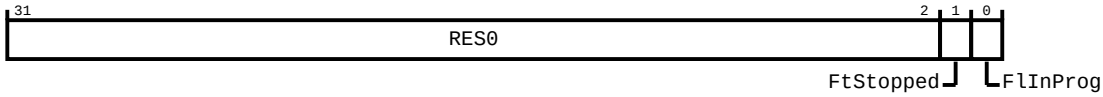
Reset value

0x0000000-

Bit descriptions

The following figure shows the FFSR register bit assignments.

Figure 9-597: Bit assignment diagram for the FFSR register



The following table shows the FFSR register bit descriptions.

Table 9-617: FFSR bit descriptions

Bits	Name	Reset	Type	Description
31:2	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
1	FtStopped	UNKNOWN	RO	Formatter Stopped. This bit behaves the same way as STS.FtEmpty.
0	FlInProg	UNKNOWN	RO	Flush In Progress. This bit indicates whether the TMC is currently processing a flush request. The flush initiation is controlled by the flush control bits in the FFCR register. This bit is cleared to 0 when leaving the Disabled state and retains its value when entering the Disabled state. When in Disabled state, this bit is not updated. **0b0** No flush activity in progress. **0b1** Flush in progress on the ATB receiver interface or the TMC internal pipeline.

9.21.8 css600_tmc_ets Formatter and Flush Control Register, FFCR

The FFCR controls the generation of stop, trigger and flush events. The insertion of a flush completion packet and the insertion of a trigger packet in the formatted trace is enabled here.

Also one of the 2 formatter modes for bypass mode and normal mode can be changed here when the formatter has stopped.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x304`

Type

RW

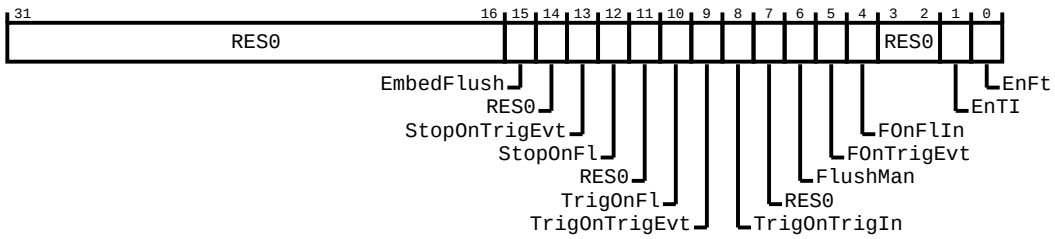
Reset value

`0x00000000`

Bit descriptions

The following figure shows the FFCR register bit assignments.

Figure 9-598: Bit assignment diagram for the FFCR register



The following table shows the FFCR register bit descriptions.

Table 9-618: FFCR bit descriptions

Bits	Name	Reset	Type	Description
31:16	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
15	EmbedFlush	0b0	RW	Embed Flush ID (flush completion packet). Enables insertion of Flush ID 0x7B with a single byte of data payload = 0x00 in the output trace, immediately after the last flush data byte, when a flush completes on the ATB receiver interface. This bit is effective only in Normal formatting modes. In Bypass mode, the Flush ID insertion remains disabled and this bit is ignored. **0b0** Disable Flush ID insertion. **0b1** Enable Flush ID insertion.
14	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
13	StopOnTrigEvt	0b0	RW	Stop On Trigger Event. If this bit is set, the formatter is stopped when a Trigger Event has been observed.
12	StopOnFl	0b0	RW	Stop On Flush. If this bit is set, the formatter is stopped on completion of a flush operation. The initiation of a flush operation is controlled by programming the register bits FFCR.FlushMan, FFCR.FOnTrigEvt, and FFCR.FOnFlIn. When a flush-initiation condition occurs, afvalid_rx is asserted, and when the flush completion is received, that is, afready_rx=1, trace capture is stopped. Any remaining data in the formatter is appended with a post-amble and written to trace memory. The flush operation is then complete.
11	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
10	TrigOnFl	0b0	RW	Indicate on trace stream the completion of flush. If this bit is set, a trigger is indicated on the trace stream when afready_rx is received for a flush in progress. If this bit is clear, no triggers are embedded in the trace stream on flush completion. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit.
9	TrigOnTrigEvt	0b0	RW	Indicate on trace stream the occurrence of a Trigger Event. If this bit is set, a trigger is indicated on the output trace stream when a Trigger Event occurs. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit.
8	TrigOnTrigIn	0b0	RW	Indicate on trace stream the occurrence of a rising edge on trigin. If this bit is set, a trigger is indicated on the trace stream when a rising edge is detected on the trigin input. If Trigger Insertion is disabled, that is, FFCR.EnTI=0, then trigger indication on the trace stream is blocked regardless of the value that is programmed in this bit.
7	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
6	FlushMan	0b0	RW	Manually generate a flush of the system. Writing 1 to this bit causes a flush to be generated. This bit is cleared automatically when, in formatter bypass mode, afready_rx was sampled high, or, in normal formatting mode, afready_rx was sampled high and all flush data was output to the trace memory. If CTL.TraceCaptEn=0, writes to this bit are ignored.
5	FOnTrigEvt	0b0	RW	Flush on Trigger Event. If FFCR.StopOnTrigEvt is set, this bit is ignored. Setting this bit generates a flush when a Trigger Event occurs. If FFCR.StopOnTrigEvt is set, this bit is ignored.
4	FOnFlIn	0b0	RW	Setting this bit enables the detection of transitions on the flushin input by the TMC. If this bit is set and the formatter has not already stopped, a rising edge on flushin initiates a flush request.
3:2	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
1	EnTI	0b0	RW	Enable Trigger Insertion. Setting this bit enables the insertion of triggers in the formatted trace stream. A trigger is indicated by inserting one byte of data 0x00 with atid_rx=0x7D in the trace stream. Trigger indication on the trace stream is also controlled by the register bits FFCR.TrigOnFl, FFCR.TrigOnTrigEvt, and FFCR.TrigOnTrigIn. This bit can only be changed when the TMC is in Disabled state. If FFCR.EnTI bit is set formatting is enabled.
0	EnFt	0b0	RW	Enable Formatter. If this bit is set, formatting is enabled. When EnTi is set, formatting is enabled. For backwards-compatibility with earlier versions of the ETB disabling of formatting is supported only in CB mode. This bit can only be changed when TMC is in Disabled state.

9.21.9 css600_tmc_ets Periodic Synchronization Counter Register, PSCR

This register determines the reload value of the Periodic Synchronization Counter. This counter enables the frequency of sync packets to be optimized to the trace capture buffer size. The default behavior of the counter is to generate periodic synchronization requests, syncreq_rx, on the ATB receiver interface.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x308`

Type

RW

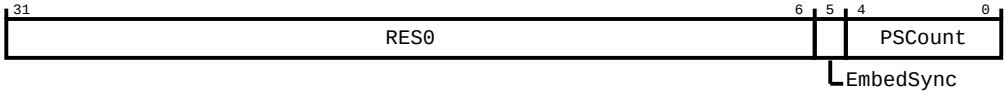
Reset value

`0x0000000A`

Bit descriptions

The following figure shows the PSCR register bit assignments.

Figure 9-599: Bit assignment diagram for the PSCR register



The following table shows the PSCR register bit descriptions.

Table 9-619: PSCR bit descriptions

Bits	Name	Reset	Type	Description
31:6	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
5	EmbedSync	`0b0`	RW	Embed Frame Sync Packet in the trace stream. Setting this bit to 1 enables the formatter to insert frame sync packets in the trace stream at periodic intervals. If this bit is set and the Synchronization Counter is enabled, the formatter inserts a 32-bit frame sync packet in the trace stream when the counter reaches 0. This bit is effective only when formatting is enabled, that is when FFCR.EnTI=1 or FFCR.EnFt=1, and it is ignored when the formatter is in bypass mode.

Bits	Name	Reset	Type	Description
4:0	PSCount	0b01010	RW	Periodic Synchronization Count. Determines the reload value of the Synchronization Counter. The reload value takes effect the next time the counter reaches zero. When trace capture is enabled, the Synchronization Counter counts the number of bytes of trace data that is stored into the trace memory, regardless of whether the trace data has been formatted by the TMC or not, since the occurrence of the last sync request on the ATB receiver interface. When the counter reaches 0, a sync request is sent on the ATB receiver interface. Reads from this register return the reload value that is programmed in this register. This field resets to 0x0A, that is, the default sync period is 2^10 bytes. If a reserved value is programmed in this register field, the value 0x1B is used instead, and subsequent reads from this register also return 0x1B. The following constraints apply to the values written to the PSCount field: 0x0 - synchronization is disabled, 0x1-0x6 - reserved, 0x7-0x1B - synchronization period is 2^PSCount bytes. The smallest value 0x7 gives a sync period of 128 bytes. The maximum allowed value 0x1B gives a sync period of 2^27 bytes, 0x1C-0x1F - reserved.

9.21.10 css600_tmc_ets Integration Test Event and Interrupt Control Register, ITEVTINTR

This register controls the values of event and interrupt outputs in integration mode. In functional mode, this register behaves as **RAZ/WI**.

In integration mode, the value that is written to any bit of this register is driven on the output pin that is controlled by that bit and the reads return 0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEE0

Type

WO

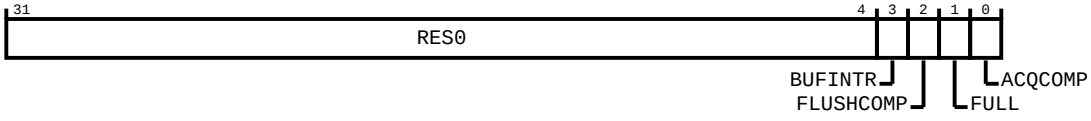
Reset value

0x00000000

Bit descriptions

The following figure shows the ITEVTINTR register bit assignments.

Figure 9-600: Bit assignment diagram for the ITEVTINTR register



The following table shows the ITEVTINTR register bit descriptions.

Table 9-620: ITEVTINTR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
3	BUFINTR	0b0	WO	Controls the value of bufintr output in integration mode.
2	FLUSHCOMP	0b0	WO	Controls the value of flushcomp output in integration mode.
1	FULL	0b0	WO	Controls the value of full output in integration mode.
0	ACQCOMP	0b0	WO	Controls the value of acqcomp output in integration mode.

9.21.11 css600_tmc_ets Integration Test Trigger In and Flush In register, ITTRFLIN

This register captures the values of the flushin and trigin inputs in integration mode. In functional mode, this register behaves as **RAZ/WI**.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEE8

Type

RO

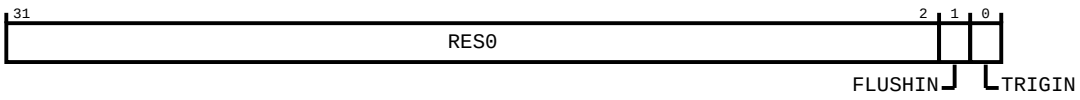
Reset value

0x00000000

Bit descriptions

The following figure shows the ITTRFLIN register bit assignments.

Figure 9-601: Bit assignment diagram for the ITTRFLIN register



The following table shows the ITTRFLIN register bit descriptions.

Table 9-621: ITTRFLIN bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	FLUSHIN	0b0	RO	Integration status of flushin input. In integration mode, this bit latches to 1 on a rising edge of the flushin input. It is cleared when the register is read or when integration mode is disabled.

Bits	Name	Reset	Type	Description
0	TRIGIN	0b0	RO	Integration status of trigin input. In integration mode, this bit latches to 1 on a rising edge of the trigin input. It is cleared when the register is read or when integration mode is disabled.

9.21.12 css600_tmc_ets Integration Test ATB Data 0 Register, ITATBDATA0

This register captures the value of atdata_rx input in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of corresponding atdata_rx bits. The width of this register is given by: 1+(ATB DATA WIDTH)/8.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEEC

Type

RO

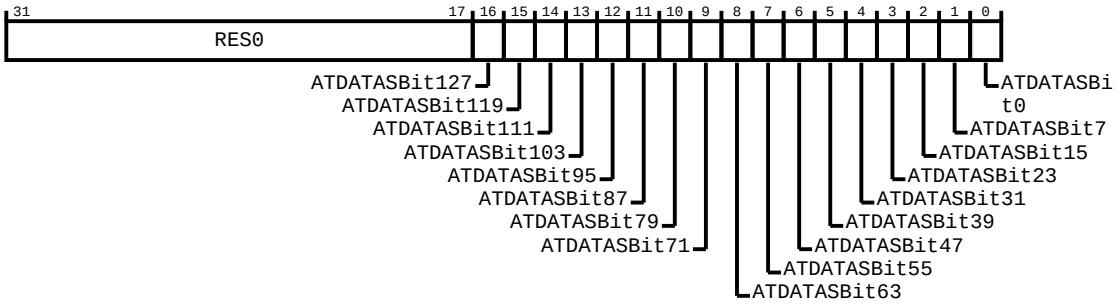
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBDATA0 register bit assignments.

Figure 9-602: Bit assignment diagram for the ITATBDATA0 register



The following table shows the ITATBDATA0 register bit descriptions.

Table 9-622: ITATBDATA0 bit descriptions

Bits	Name	Reset	Type	Description
31:17	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
16	ATDATASBit127	0b0	RO	Returns the value of atdata_rx[127] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
15	ATDATASBit119	0b0	RO	Returns the value of atdata_rx[119] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
14	ATDATASBit111	0b0	RO	Returns the value of atdata_rx[111] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
13	ATDATASBit103	0b0	RO	Returns the value of atdata_rx[103] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
12	ATDATASBit95	0b0	RO	Returns the value of atdata_rx[95] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
11	ATDATASBit87	0b0	RO	Returns the value of atdata_rx[87] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
10	ATDATASBit79	0b0	RO	Returns the value of atdata_rx[79] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
9	ATDATASBit71	0b0	RO	Returns the value of atdata_rx[71] input in integration mode. **RES0** if ATB_DATA_WIDTH < 128.
8	ATDATASBit63	0b0	RO	Returns the value of atdata_rx[63] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
7	ATDATASBit55	0b0	RO	Returns the value of atdata_rx[55] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
6	ATDATASBit47	0b0	RO	Returns the value of atdata_rx[47] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
5	ATDATASBit39	0b0	RO	Returns the value of atdata_rx[39] input in integration mode. **RES0** if ATB_DATA_WIDTH < 64.
4	ATDATASBit31	0b0	RO	Returns the value of atdata_rx[31] input in integration mode.
3	ATDATASBit23	0b0	RO	Returns the value of atdata_rx[23] input in integration mode.
2	ATDATASBit15	0b0	RO	Returns the value of atdata_rx[15] input in integration mode.
1	ATDATASBit7	0b0	RO	Returns the value of atdata_rx[7] input in integration mode.
0	ATDATASBit0	0b0	RO	Returns the value of atdata_rx[0] input in integration mode.

9.21.13 css600_tmc_ets Integration Test ATB Control 2 Register, ITATBCTR2

This register enables control of ATB receiver outputs atready_rx, afvalid_rx, and syncreq_rx in integration mode. In functional mode, this register behaves as **RAZ/WI**.

In integration mode, the value that is written to any bit of this register is driven on the output pin that is controlled by that bit and the reads return 0x0.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEF0`

Type

WO

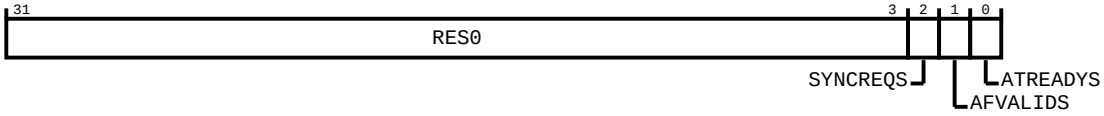
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITATBCTR2 register bit assignments.

Figure 9-603: Bit assignment diagram for the ITATBCTR2 register



The following table shows the ITATBCTR2 register bit descriptions.

Table 9-623: ITATBCTR2 bit descriptions

Bits	Name	Reset	Type	Description
31:3	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
2	SYNCREQS	0b0	WO	Controls the value of syncreq_rx output in integration mode.
1	AFVALIDS	0b0	WO	Controls the value of afvalid_rx output in integration mode.
0	ATREADYS	0b0	WO	Controls the value of atready_rx output in integration mode.

9.21.14 css600_tmc_ets Integration Test ATB Control 1 Register, ITATBCTR1

This register captures the value of the atid_rx[6:0] input in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of atid_rx input.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF4

Type

RO

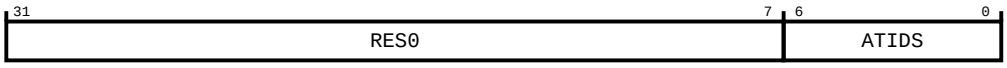
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR1 register bit assignments.

Figure 9-604: Bit assignment diagram for the ITATBCTR1 register



The following table shows the ITATBCTR1 register bit descriptions.

Table 9-624: ITATBCTR1 bit descriptions

Bits	Name	Reset	Type	Description
31:7	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
6:0	ATIDS	0x0	RO	Returns the value of atid_rx[6:0] input in integration mode.

9.21.15 css600_tmc_ets Integration Test ATB Control 0 Register, ITATBCTR0

This register captures the values of ATB receiver inputs atvalid_rx, afready_rx, atwakeup_rx, and atbytes_rx in integration mode. In functional mode, this register behaves as **RAZ/WI**. In integration mode, writes to this register are ignored and the reads return the value of corresponding input pins.

The width of this register is given by: 8+log2(ATB DATA WIDTH/8)..

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF8

Type

RO

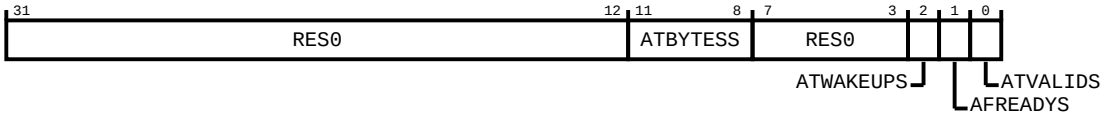
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR0 register bit assignments.

Figure 9-605: Bit assignment diagram for the ITATBCTR0 register



The following table shows the ITATBCTR0 register bit descriptions.

Table 9-625: ITATBCTR0 bit descriptions

Bits	Name	Reset	Type	Description
31:12	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
11:8	ATBYTESS	0b0000	RO	Returns the value of atbytes_rx input in integration mode. N=8+log2(ATB DATA WIDTH/8).

Bits	Name	Reset	Type	Description
7:3	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
2	ATWAKEUPS	0b0	RO	Returns the value of atwakeup_rx input in integration mode.
1	AFREADYS	0b0	RO	Returns the value of afready_rx input in integration mode.
0	ATVALIDS	0b0	RO	Returns the value of atvalid_rx input in integration mode.

9.21.16 css600_tmc_ets Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

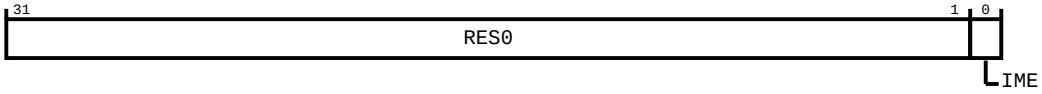
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-606: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-626: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.21.17 css600_tmc_ets Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

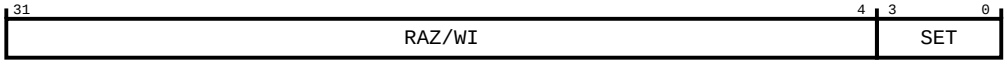
Reset value

0x0000000F

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-607: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-627: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:4	RAZ/WI	0x0	RO	RAZ/WI.
3:0	SET	0b1111	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.21.18 css600_tmc_ets Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFA4`

Type

RW

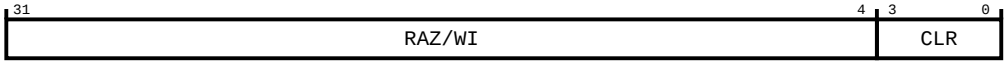
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-608: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-628: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	`0x0`	RO	**RAZ/WI**.
3:0	CLR	`0b0000`	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.21.19 css600_tmc_ets Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

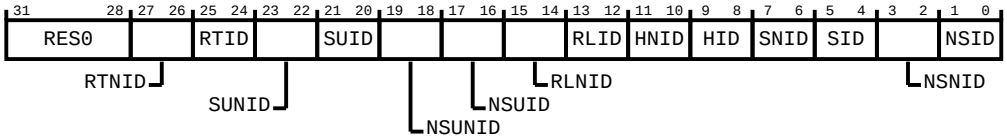
Reset value

0x00000000

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-609: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-629: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	0b00	RO	Root non-invasive debug. **0b00** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.
25:24	RTID	0b00	RO	Root invasive debug. **0b00** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.

Bits	Name	Reset	Type	Description
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	0b00	RO	Secure non-invasive debug. **0b00** Debug level is not supported.
5:4	SID	0b00	RO	Secure invasive debug. **0b00** Debug level is not supported.
3:2	NSNID	0b00	RO	Non-secure non-invasive debug. **0b00** Debug level is not supported.
1:0	NSID	0b00	RO	Non-secure invasive debug. **0b00** Debug level is not supported.

9.21.20 css600_tmc_ets Device Configuration Register 1, DEVID1

Contains an **IMPLEMENTATION DEFINED** value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC4

Type

RO

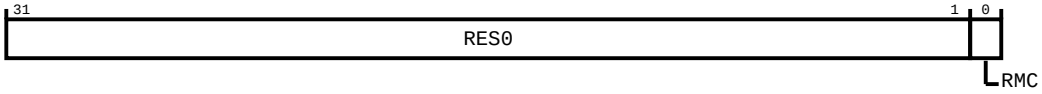
Reset value

0x00000001

Bit descriptions

The following figure shows the DEVID1 register bit assignments.

Figure 9-610: Bit assignment diagram for the DEVID1 register



The following table shows the DEVID1 register bit descriptions.

Table 9-630: DEVID1 bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	RMC	0b1	RO	Register management mode. TMC implements register management mode 1.

9.21.21 css600_tmc_ets Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFC8

Type

RO

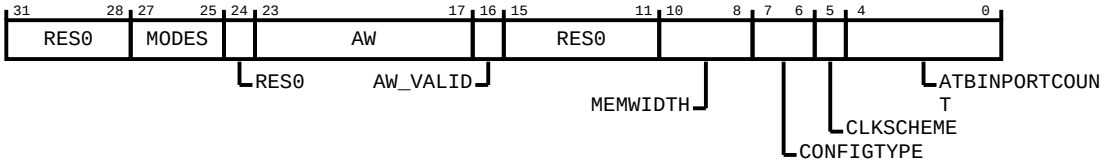
Reset value

0x04010-C0

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-611: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-631: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:25	MODES	0b010	RO	Indicates the supported modes of operation. Reads as 3'b010 indicating that ETS supports CB mode.
24	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
23:17	AW	0x0	RO	This field indicates the width of AXI address bus in ETR configuration. This field is valid only when DEVID.AW_VALID is set. Since AXI-Stream doesn't have an address bus, this field reads as 0x00.
16	AW_VALID	0b1	RO	Indicates whether field DEVID.AW is valid. The value of this field is fixed at 1.
15:11	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
10:8	MEMWIDTH	**IMPLEMENTATION DEFINED**	RO	Indicates the width of the internal memory data bus. For the ETS this value is equal to ATB_DATA_WIDTH. **0b010** Memory interface databus is 32 bits wide. (ATB_DATA_WIDTH = 32bit) **0b011** Memory interface databus is 64 bits wide. (ATB_DATA_WIDTH = 64bit) **0b100** Memory interface databus is 128 bits wide. (ATB_DATA_WIDTH = 128bit)
7:6	CONFIGTYPE	0b11	RO	Indicates the TMC configuration. **0b11** ETS - Embedded Trace Streamer
5	CLKSCHEME	0b0	RO	RAM Clocking Scheme. This value indicates the TMC RAM clocking scheme used, that is, whether the TMC RAM operates synchronously or asynchronously to the TMC clock. Fixed to 0 indicating that TMC RAM clock is synchronous to the clk input.
4:0	ATBINPORTCOUNT	0b00000	RO	Hidden Level of ATB input multiplexing. This value indicates the type/number of ATB multiplexing present on the input ATB. Fixed to 0x00 indicating that no multiplexing is present.

9.21.22 css600_tmc_ets Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFCC

Type

RO

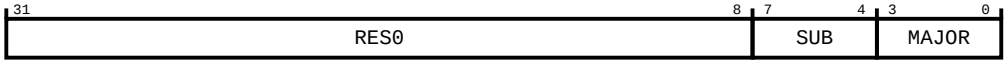
Reset value

`0x00000021`

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-612: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-632: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	`0b0010`	RO	Minor classification. Returns `0x2`, indicating this component is a Buffer.
3:0	MAJOR	`0b0001`	RO	Major classification. Returns `0x1`, indicating this component is a Trace Sink.

9.21.23 css600_tmc_ets Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD0`

Type

RO

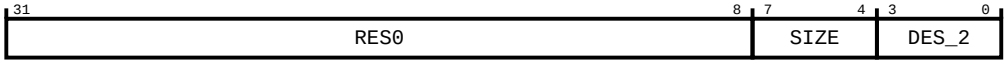
Reset value

`0x00000004`

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-613: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-633: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.21.24 css600_tmc_ets Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

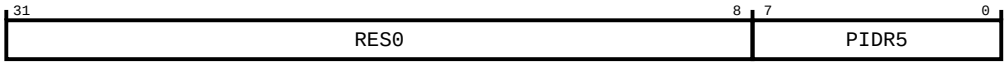
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-614: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-634: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.21.25 css600_tmc_ets Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD8

Type

RO

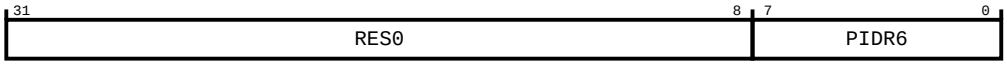
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-615: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-635: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.21.26 css600_tmc_ets Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

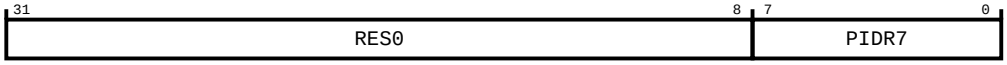
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-616: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-636: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.21.27 css600_tmc_ets Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

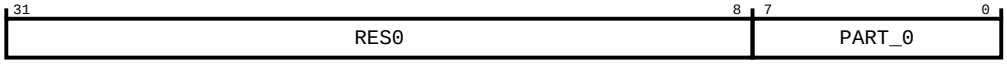
Reset value

0x000000E8

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-617: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-637: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xE8	RO	Part number (lower 8 bits). **0xE8** ETR or ETS - Embedded Trace Router or Streamer

9.21.28 css600_tmc_ets Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

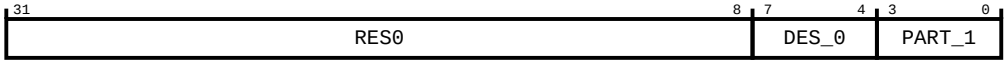
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-618: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-638: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.21.29 css600_tmc_ets Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

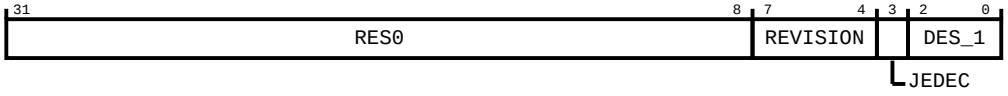
Reset value

0x0000009B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-619: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-639: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b1001	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.

Bits	Name	Reset	Type	Description
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.21.30 css600_tmc_ets Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFEC

Type

RO

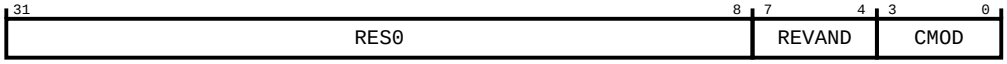
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-620: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-640: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.21.31 css600_tmc_ets Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF0`

Type

RO

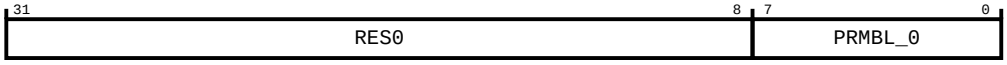
Reset value

`0x0000000D`

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-621: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-641: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	`0xD`	RO	Preamble. Returns `0x0D`.

9.21.32 css600_tmc_ets Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF4`

Type

RO

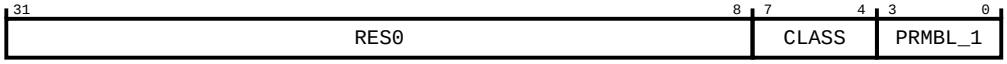
Reset value

`0x00000090`

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-622: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-642: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	`0b1001`	RO	Component class **`0b1001`** CoreSight component
3:0	PRMBL_1	`0b0000`	RO	Preamble. Returns `0x0`.

9.21.33 css600_tmc_ets Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF8`

Type

RO

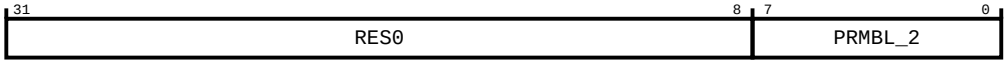
Reset value

`0x00000005`

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-623: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-643: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.21.34 css600_tmc_ets Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

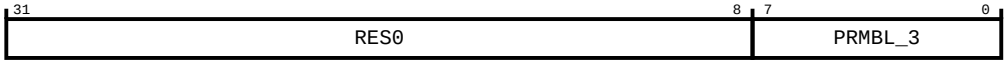
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-624: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-644: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.22 css600_tpiu register summary

This section describes the css600_tpiu_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-645: css600_tpiu_registers register summary

Offset	Name	Type	Reset	Width	Description
0x000	SSPSR	RO	0x--------	32-bit	Supported Port Size Register
0x004	CSPSR	RW	0x00000001	32-bit	Current Port Size Register
0x100	STMR	RO	0x0000011F	32-bit	Supported Trigger Modes Register
0x104	TCVR	RW	0x00000000	32-bit	Trigger Counter Value Register
0x108	TCMR	RW	0x00000000	32-bit	Trigger Counter Multiplier Register
0x200	STPMR	RO	0x0003000F	32-bit	Supported Test Patterns/Modes Register
0x204	CTPMR	RW	0x00000000	32-bit	Current Test Patterns/Modes Register
0x208	TPRCR	RW	0x00000000	32-bit	Test Pattern Repeat Counter Register
0x300	FFSR	RO	0x0000000-	32-bit	Formatter and Flush Status Register
0x304	FFCR	RW	0x00001000	32-bit	Formatter and Flush Control Register
0x308	FSCR	RW	0x00000040	32-bit	Formatter Synchronization Count Register
0x400	EXTCTLIN	RO	0x000000--	32-bit	External Control Port In Register
0x404	EXTCTLOUT	RW	0x00000000	32-bit	External Control Port Out Register
0xee8	ITTRFLIN	RO	0x00000000	32-bit	Integration Test Trigger In and Flush In Register
0xeec	ITATBDATA0	RO	0x00000000	32-bit	Integration Test ATB Data Register 0
0xef0	ITATBCTR2	WO	0x00000000	32-bit	Integration Test ATB Control Register 2
0xef4	ITATBCTR1	RO	0x00000000	32-bit	Integration Test ATB Control Register 1
0xef8	ITATBCTR0	RO	0x00000000	32-bit	Integration Test ATB Control Register 0
0xefc	ITOUTCTR	WO	0x00000000	32-bit	Integration Test Output Control Register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfa0	CLAIMSET	RW	0x0000000F	32-bit	Claim Tag Set Register
0xfa4	CLAIMCLR	RW	0x00000000	32-bit	Claim Tag Clear Register
0xfb8	AUTHSTATUS	RO	0x00000000	32-bit	Authentication Status Register
0xfbc	DEVARCH	RO	0x00000000	32-bit	Device Architecture Register
0xfc8	DEVID	RO	0x00000020	32-bit	Device Configuration Register
0xfcc	DEVTYPE	RO	0x00000011	32-bit	Device Type Identifier Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x000000E7	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B9	32-bit	Peripheral Identification Register 1

Offset	Name	Type	Reset	Width	Description
0xfe8	PIDR2	RO	0x0000004B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x00000090	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.22.1 css600_tpiu Supported Port Size Register, SSPSR

The SSPSR register shows supported width configurations of the tracedata port. Each bit location represents a single port size that is supported, that is sizes 32 bits down to 1 bit, in bit locations [31:0]. If a bit is set, then that port size is supported.

By default, the RTL is designed to support all port sizes. Port sizes, other than 1-bit, are configuration-dependent on the tie-off value of tp_maxdatasize. Bit[0] is always 1.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x000

Type

RO

Reset value

0x--------

Bit descriptions

The following figure shows the SSPSR register bit assignments.

Figure 9-625: Bit assignment diagram for the SSPSR register

The following table shows the SSPSR register bit descriptions.

Table 9-646: SSPSR bit descriptions

Bits	Name	Reset	Type	Description
31:0	SSPSR	IMPLEMENTATION DEFINED	RO	Supported tracedata port sizes. Bit[0] is always 1.

9.22.2 css600_tpiu Current Port Size Register, CSPSR

The CSPSR register shows the currently selected size of the tracedata port. It has the same format as the Supported Port Size Register but only one bit is set to show the currently selected port size.

If a bit that is indicated as not supported in the SSPSR is set in the CSPSR, it can corrupt the output trace stream, in trace capture mode, and the trace patterns in pattern generation mode. If more than one bit is set, this register indicates the programmed size. However, the port size is internally resolved to the highest order set bit. This register must not be modified while the trace port is still active, or without correctly stopping the formatter. If this happens, it can result in data not being aligned to the port width, for example, data on an 8-bit trace port might not be byte aligned. For the register access to complete on APB clocking on traceclkin is needed.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x004`

Type

RW

Reset value

`0x00000001`

Bit descriptions

The following figure shows the CSPSR register bit assignments.

Figure 9-626: Bit assignment diagram for the CSPSR register

The following table shows the CSPSR register bit descriptions.

Table 9-647: CSPSR bit descriptions

Bits	Name	Reset	Type	Description
31:0	CSPSR	`0x1`	RW	Currently selected size of the tracedata port

9.22.3 css600_tpiu Supported Trigger Modes Register, STMR

The STMR register indicates the implemented Trigger Counter multipliers and other supported features of the trigger system.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x100`

Type

RO

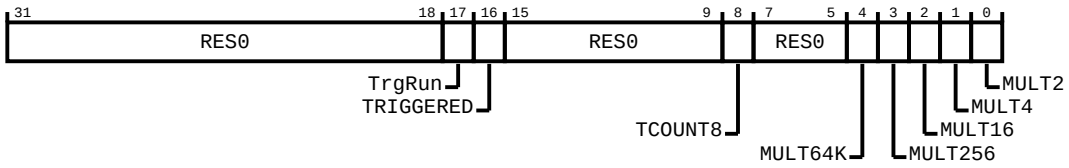
Reset value

`0x0000011F`

Bit descriptions

The following figure shows the STMR register bit assignments.

Figure 9-627: Bit assignment diagram for the STMR register



The following table shows the STMR register bit descriptions.

Table 9-648: STMR bit descriptions

Bits	Name	Reset	Type	Description
31:18	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
17	TrgRun	`0b0`	RO	A trigger has occurred after a rising edge of trigin input, or a trigger packet (atid_rx = `0x7D`) is received in the input trace: **`0b0`** Either a trigger has not occurred or the counter is at 0 **`0b1`** A trigger has occurred but the counter is not at 0

Bits	Name	Reset	Type	Description
16	TRIGGERED	0b0	RO	A trigger has occurred after a rising edge of trigin input, or a trigger packet (atid_rx = 0x7D) is received in the input trace and the counter has reached 0. This bit is cleared when the TCVR or TCMR register is written. **0b0** Trigger has not occurred **0b1** Trigger has occurred
15:9	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
8	TCOUNT8	0b1	RO	Returns 1 indicating that an 8-bit wide counter register is implemented for trigger insertion.
7:5	**RES0**	0b000	RO	Reserved bit or field with SBZP behavior.
4	MULT64K	0b1	RO	Returns 1, indicating that multiplying the trigger counter by 65536 is supported
3	MULT256	0b1	RO	Returns 1, indicating that multiplying the trigger counter by 256 is supported
2	MULT16	0b1	RO	Returns 1, indicating that multiplying the trigger counter by 16 is supported
1	MULT4	0b1	RO	Returns 1, indicating that multiplying the trigger counter by 4 is supported
0	MULT2	0b1	RO	Returns 1, indicating that multiplying the trigger counter by 2 is supported

9.22.4 css600_tpiu Trigger Counter Value Register, TCVR

The TCVR register indicates the programmed trigger counter value. The TPIU implements a trigger counter that enables delaying the indication of triggers to any external connected trace capture devices.

The counter is 8 bits wide and is intended only to be used with the counter multipliers within the Trigger Multiplier Register. When a trigger occurs (observed trigin=1 or atid_rx=0x7D), the TCVR.TrigCount value, with the TCMR, determines the number of words to be output from the formatter before the trigger is indicated on the Trace Port. When the trigger counter reaches zero, the value from the TCVR register is reloaded into the trigger counter. Writing to this register causes the trigger counter (the actual counter) to be reloaded. Reading this register returns the programmed count value and not the current count.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x104

Type

RW

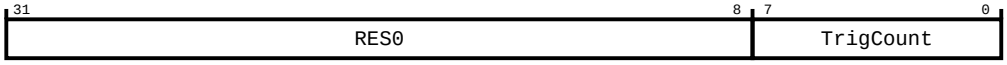
Reset value

0x00000000

Bit descriptions

The following figure shows the TCVR register bit assignments.

Figure 9-628: Bit assignment diagram for the TCVR register



The following table shows the TCVR register bit descriptions.

Table 9-649: TCVR bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	TrigCount	0x0	RW	Programmed trigger counter value.

9.22.5 css600_tpiu Trigger Counter Multiplier Register, TCMR

The TCMR register contains the selectors for the trigger counter multiplier. Several multipliers can be selected to create the required multiplier value between 1 (TCMR[4:0] = 0x0) and 2^31 (TCMR[4:0] = 0x1F). When more than one bit it set, the effective multiplier value is the product of selected multipliers.

Writing to this register causes the trigger counter (the actual counter) to be reloaded and the state in the multipliers to be reset.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x108

Type

RW

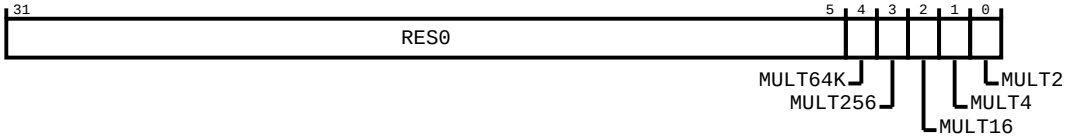
Reset value

0x00000000

Bit descriptions

The following figure shows the TCMR register bit assignments.

Figure 9-629: Bit assignment diagram for the TCMR register



The following table shows the TCMR register bit descriptions.

Table 9-650: TCMR bit descriptions

Bits	Name	Reset	Type	Description
31:5	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
4	MULT64K	0b0	RW	Multiply the Trigger Counter by 65536: **0b0** Multiplier disabled **0b1** Multiplier enabled
3	MULT256	0b0	RW	Multiply the Trigger Counter by 256: **0b0** Multiplier disabled **0b1** Multiplier enabled
2	MULT16	0b0	RW	Multiply the Trigger Counter by 16: **0b0** Multiplier disabled **0b1** Multiplier enabled
1	MULT4	0b0	RW	Multiply the Trigger Counter by 4: **0b0** Multiplier disabled **0b1** Multiplier enabled
0	MULT2	0b0	RW	Multiply the Trigger Counter by 2: **0b0** Multiplier disabled **0b1** Multiplier enabled

9.22.6 css600_tpiu Supported Test Patterns/Modes Register, STPMR

The STPMR Register provides a set of known bit sequences or patterns that can be output over the trace port and can be detected by the TPA or TCD.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x200`

Type

RO

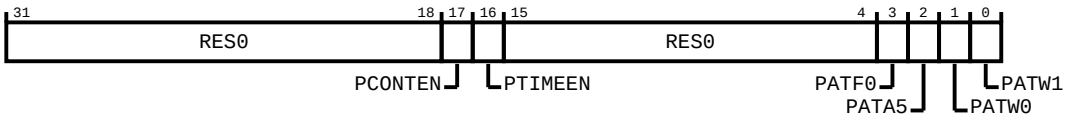
Reset value

`0x0003000F`

Bit descriptions

The following figure shows the STPMR register bit assignments.

Figure 9-630: Bit assignment diagram for the STPMR register



The following table shows the STPMR register bit descriptions.

Table 9-651: STPMR bit descriptions

Bits	Name	Reset	Type	Description
31:18	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
17	PCONTEN	`0b1`	RO	Continuous Pattern Mode, returns 1 indicating that continuous pattern mode is supported
16	PTIMEEN	`0b1`	RO	Timed Pattern Mode, returns 1 indicating that timed pattern mode is supported
15:4	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
3	PATF0	`0b1`	RO	FF/00 Pattern, returns 1 indicating that the FF/00 pattern is supported over the trace port
2	PATA5	`0b1`	RO	55/AA Pattern, returns 1 indicating that the 55/AA pattern is supported over the trace port
1	PATW0	`0b1`	RO	Walking 0 Pattern, returns 1 indicating that the walking 0s pattern is supported over the trace port
0	PATW1	`0b1`	RO	Walking 1s Pattern, returns 1 indicating that the walking 1s pattern is supported over the trace port

9.22.7 css600_tpiu Current Test Patterns/Modes Register, CTPMR

The CTPMR register indicates the current test pattern or mode selected. Only one of the two mode bits, bits[17:16], can be set at any one time, but a multiple number of bits for the patterns can be set using bits [3:0].

When timed mode is selected, after the allotted number of cycles is reached, the mode automatically switches to off mode. The pattern with higher bit index is output first when multiple patterns are selected. When no pattern is selected then a default pattern (00/00) is used instead. In continuous mode, the pattern generator continues to send patterns until CTPMR.PCONTEN bit is cleared by software. If multiple patterns are enabled, after sending out all enabled patterns, the pattern generator switches back to the first pattern type and continues to do so until stopped by software. When no pattern is selected then the default pattern (00/00) is used instead. Writing to this register when timed or continuous pattern mode is already enabled causes the current pattern generation to be abandoned and to be restarted with the new pattern mode and new pattern set. Writing to TPRCR or CSPSR when timed or continuous pattern mode is already enabled causes the current pattern generation to be abandoned and to be restarted. The reset value of this register is 0x00000000 which indicates off mode with no selected patterns. For the register access to complete on APB clocking on traceclkin is needed.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x204`

Type

RW

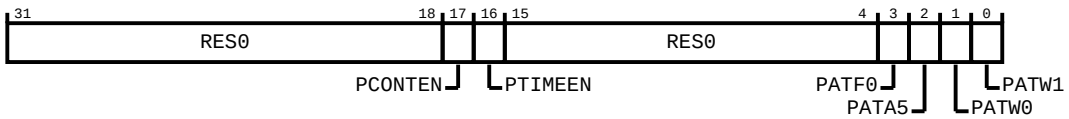
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CTPMR register bit assignments.

Figure 9-631: Bit assignment diagram for the CTPMR register



The following table shows the CTPMR register bit descriptions.

Table 9-652: CTPMR bit descriptions

Bits	Name	Reset	Type	Description
31:18	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
17	PCONTEN	0b0	RW	Continuous Pattern Mode. Indicates whether continuous pattern mode is enabled: **0b0** Mode disabled **0b1** Mode enabled
16	PTIMEEN	0b0	RW	Timed Pattern Mode. Indicates whether timed pattern mode is enabled: **0b0** Mode disabled **0b1** Mode enabled
15:4	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
3	PATF0	0b0	RW	FF/00 Pattern. Indicates whether the FF/00 pattern is enabled as output over the Trace Port. All pins toggle simultaneously as 1-0-1-0. **0b0** Pattern disabled **0b1** Pattern enabled
2	PATA5	0b0	RW	55/AA Pattern. Indicates whether the 55/AApattern is enabled as output over the Trace Port. The odd numbered pins toggle as 0-1-0-1, while the even numbered pins toggle as 1-0-1-0, simultaneously. **0b0** Pattern disabled **0b1** Pattern enabled
1	PATW0	0b0	RW	Walking 0 Pattern. Indicates whether the walking 0s pattern is enabled as output over the Trace port. To start with, all pins are set to 1, except tracedata[0] which is driven LOW. In each subsequent cycle, the 0 bit shifts to its left by 1 position and eventually rotates around from its starting position, based on the CSPSR value, to tracedata[0], provided the pattern mode remains enabled for a sufficient number of cycles. When timed mode is selected, after the allotted number of cycles is reached (See TPRCR, 0x208), the wsmode automatically switches to off mode. The pattern with higher bit index is output first when multiple patterns are selected. **0b0** Pattern disabled **0b1** Pattern enabled
0	PATW1	0b0	RW	Walking 1s Pattern. Indicates whether the walking 1s pattern is enabled as output over the Trace Port. It is similar to the walking 0s pattern except that tracedata[0] is set to 1 to start with and all other bits are 0. It is this set bit that rotates through all the selected pins of the tracedata port. **0b0** Pattern disabled **0b1** Pattern enabled

9.22.8 css600_tpiu Test Pattern Repeat Counter Register, TPRCR

This register indicates the number of times each test pattern is output on the Trace Port before switching to next pattern. For the register access to complete on APB clocking on traceclkin is needed.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x208`

Type

RW

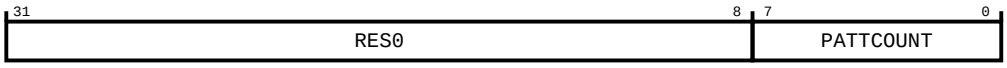
Reset value

`0x00000000`

Bit descriptions

The following figure shows the TPRCR register bit assignments.

Figure 9-632: Bit assignment diagram for the TPRCR register



The following table shows the TPRCR register bit descriptions.

Table 9-653: TPRCR bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:0	PATTCOUNT	`0x0`	RW	An 8-bit counter value that indicates the number of traceclk cycles for which a pattern runs before it switches to the next enabled pattern. A write sets the initial counter value, and a read returns the programmed value. The pattern length is PATTCOUNT+1. The reset value is `0x0`.

9.22.9 css600_tpiu Formatter and Flush Status Register, FFSR

The FFSR indicates the current status of formatter and flush features available in the TPIU.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x300

Type

RO

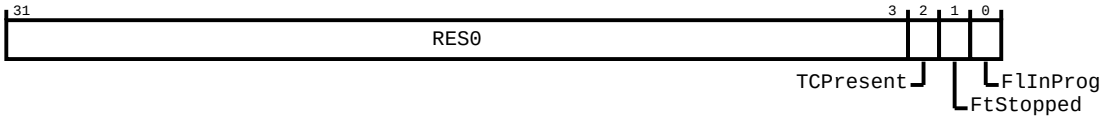
Reset value

0x0000000-

Bit descriptions

The following figure shows the FFSR register bit assignments.

Figure 9-633: Bit assignment diagram for the FFSR register



The following table shows the FFSR register bit descriptions.

Table 9-654: FFSR bit descriptions

Bits	Name	Reset	Type	Description
31:3	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
2	TCPresent	IMPLEMENTATION DEFINED	RO	Indicates whether the tracectl pin is available for use, based on the tie-off value of tpctl_valid. **0b0** tracectl pin not present. The data formatter must be used and in continuous mode. **0b1** tracectl pin present.
1	FtStopped	0b1	RO	The formatter has received a stop request and all trace data and post-amble is sent. Any further trace on the ATB interface is dropped and atready_rx is asserted. **0b0** Formatter running. **0b1** Formatter stopped.
0	FlInProg	0b0	RO	Indicates whether a flush is in progress. It is set when the TPIU sends a flush request on its ATB receiver interface. The bit remains set until the ATB flush is complete and the last byte of flush data, including the flush ID payload if FFCR.EmbedFlush is set, has been output on the trace port. **0b0** No ongoing flush. **0b1** Flush in progress.

9.22.10 css600_tpiu Formatter and Flush Control Register, FFCR

The FFCR controls the generation of stop, trigger and flush events. The insertion of a flush completion packet and the insertion of a trigger packet in the formatted trace is enabled here. In bypass mode formatting is disabled and triggers are indicated on tracectl pin, bits[1:0] must be 0b00.

In normal mode formatting is enabled and triggers are indicated on tracectl pin, bits[1:0] must be 0b01. In continuous mode formatting is enabled and triggers are embedded in the trace stream with Triger Byte ID 0x7D with a single byte of data payload = 0x00, bits[1:0] must be 0b10. Setting both bits is the same as setting bit[1]. All three flush-generating conditions can be enabled together. However, if a second or third flush event is generated from another condition then the current flush completes before the next flush is serviced. Flush from flushin takes priority over flush from trigger, which in turn completes before a manually activated flush. All trigger indication conditions can be enabled simultaneously although this can cause the appearance of multiple triggers if flush using trigger is also enabled. Both Stop On settings can be enabled although if Flush on Trigger is set up, none of the flushed data is stored. ARM recommends that you change the trace port width without enabling continuous mode. Enabling continuous mode causes data to be sent from the trace port and modifying the port size can result in data not being aligned.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x304`

Type

RW

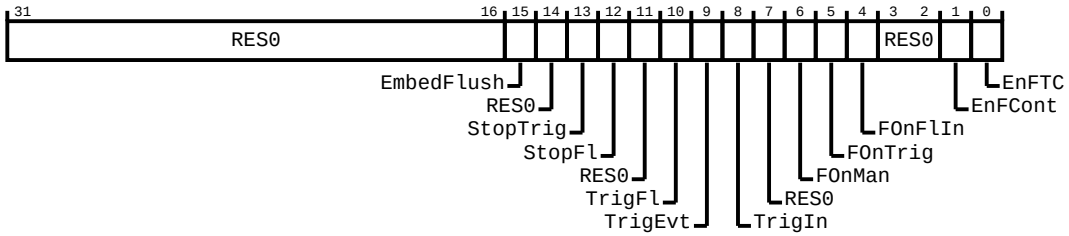
Reset value

`0x00001000`

Bit descriptions

The following figure shows the FFCR register bit assignments.

Figure 9-634: Bit assignment diagram for the FFCR register



The following table shows the FFCR register bit descriptions.

Table 9-655: FFCR bit descriptions

Bits	Name	Reset	Type	Description
31:16	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
15	EmbedFlush	0b0	RW	Embed flush completion packet, Flush ID. Enables insertion of Flush ID 0x7B with a single byte of data payload = 0x00 in the output trace, after the last flush data byte, when a flush completes on the ATB receiver interface. This bit is effective only in Normal and Continuous formatter modes. In Bypass mode, the Flush ID insertion remains disabled and this bit is ignored. **0b0** Disable Flush ID insertion **0b1** Enable Flush ID insertion
14	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
13	StopTrig	0b0	RW	Stop on Trigger Event. Stops the formatter after a trigger event is observed. **0b0** Disable stopping the formatter after a trigger event is observed. **0b1** Enable stopping the formatter after a trigger event is observed.
12	StopFl	0b1	RW	Stop on Flush Completion. Forces the FIFO to drain off any partially completed packets after a flush completion and stops the formatter. **0b0** Disable stopping the formatter when afready_rx is received. **0b1** Enable stopping the formatter when afready_rx is received.
11	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
10	TrigFl	0b0	RW	Trigger on Flush Completion. **0b0** Disable trigger indication on flush completion, that is, when afready_rx is high. **0b1** Enable trigger indication on flush completion.
9	TrigEvt	0b0	RW	Trigger on Trigger Event. Indicates a trigger when the trigger counter reaches 0 while decrementing. If FFCR.StopTrig is set, this bit is ignored. **0b0** Disable trigger indication on trigger event. **0b1** Enable trigger indication on trigger event.
8	TrigIn	0b0	RW	Trigger on trigin. **0b0** Disable trigger indication when trigin is asserted. **0b1** Enable trigger indication when trigin is asserted.
7	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.
6	FOnMan	0b0	RW	Flush Manual. Writing 1 to this bit generates a flush request on afvalid_rx pin, writing 0 has no effect. It is automatically cleared when the generated flush request completes and afready_rx is received by the TPIU. Reading this bit returns its current value.

Bits	Name	Reset	Type	Description
5	FOnTrig	0b0	RW	Flush on Trigger Event. Initiates a flush request when a trigger event occurs. A trigger event occurs when the trigger counter reaches 0 while decrementing, or, if the TCVR is 0x0 and trigin goes HIGH. **0b0** Disable generation of flush when a trigger event occurs. **0b1** Enable generation of flush when a trigger event occurs.
4	FOnFlIn	0b0	RW	Flush on flushin. **0b0** Disable generation of flush using flushin input. **0b1** Enable generation of flush using flushin input.
3:2	**RES0**	0b00	RO	Reserved bit or field with SBZP behavior.
1	EnFCont	0b0	RW	Enable Continuous Formatting Mode and Enable Formatter. The trigger packets are embedded in the trace stream. This bit can only be changed when FFSR.FtStopped is HIGH.
0	EnFTC	0b0	RW	Enable Formatter. The trigger packets are not embedded in the trace stream and the trace disable cycles and triggers are indicated by tracectl pin where present. This bit can only be changed when FFSR.FtStopped is HIGH.

9.22.11 css600_tpiu Formatter Synchronization Count Register, FSCR

The FSCR register indicates the maximum number of formatter frames sent to Trace Port after which a synchronization packet must be inserted. The register value indicates the programmed counter value and not the current state of the counter.

The TPIU uses a frame sync counter that contains the number of formatter frames since the last frame synchronization packet. The counter is a 12-bit counter with a maximum count value of 4096. This equates to synchronization every 65536 bytes (4096 packets x 16 bytes per packet). On reset, the FSCR is set up for a synchronization packet every 1024 bytes, that is every 64 formatter frames. If the formatter is configured in continuous mode, full and half-word sync frames are inserted during normal operation. In this case, the counter value is the maximum number of complete frames between full synchronization packets. For the register access to complete on APB clocking on traceclkin is needed.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x308

Type

RW

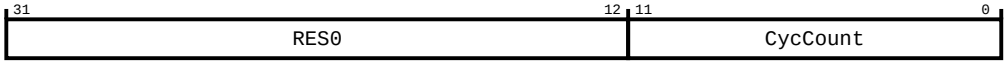
Reset value

0x00000040

Bit descriptions

The following figure shows the FSCR register bit assignments.

Figure 9-635: Bit assignment diagram for the FSCR register



The following table shows the FSCR register bit descriptions.

Table 9-656: FSCR bit descriptions

Bits	Name	Reset	Type	Description
31:12	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
11:0	CycCount	0x40	RW	12-bit counter value to indicate the number of complete frames between full synchronization packets. It is also used to send periodic synchronization requests to the ATB transmitter using syncreq_rx output. If this field is programmed as 0x0, the synchronization counter is disabled. If this field is programmed with 0x1-0x7 the programmed value is 0x8. The reset value is 0x040, that is, 64 frames = 1024 bytes.

9.22.12 css600_tpiu External Control Port In Register, EXTCTLIN

Indicates the current status of external control input port extctl_in[7:0]. It can be used as a feedback mechanism for any serializers, pin sharing multiplexers, or other solutions that might be added to the trace output pins either for pin control or a high-speed trace port solution.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x400

Type

RO

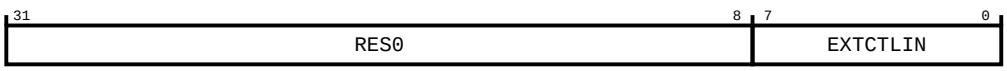
Reset value

0x000000--

Bit descriptions

The following figure shows the EXTCTLIN register bit assignments.

Figure 9-636: Bit assignment diagram for the EXTCTLIN register



The following table shows the EXTCTLIN register bit descriptions.

Table 9-657: EXTCTLIN bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	EXTCTLIN	**UNKNOWN**	RO	This 8-bit field shows the current status of external control input port extctl_in[7:0]. The reset value depends on the external source driving this port.

9.22.13 css600_tpiu External Control Port Out Register, EXTCTLOUT

Value to be driven on external control output port extctl_out[7:0]. It can be used as a control mechanism for any serializers, pin sharing multiplexers, or other solutions that might be added to the trace output pins either for pin control or a high-speed trace port solution.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x404

Type

RW

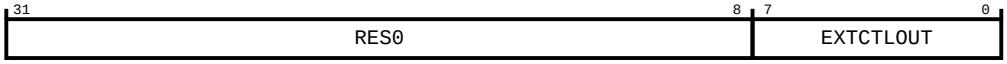
Reset value

0x00000000

Bit descriptions

The following figure shows the EXTCTLOUT register bit assignments.

Figure 9-637: Bit assignment diagram for the EXTCTLOUT register



The following table shows the EXTCTLOUT register bit descriptions.

Table 9-658: EXTCTLOUT bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	EXTCTLOUT	0x0	RW	This 8-bit field holds the value to be driven on the external control output port extctl_out[7:0].

9.22.14 css600_tpiu Integration Test Trigger In and Flush In Register, ITTRFLIN

This register indicates the integration status of the flushin and trigin inputs in integration mode. Reads are allowed even in functional mode, but the register itself is disabled and does not get updated even if the inputs change. The reset value depends on the external source driving the inputs.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEE8`

Type

RO

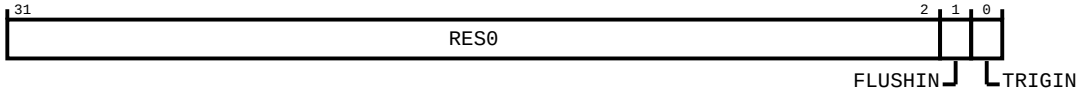
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITTRFLIN register bit assignments.

Figure 9-638: Bit assignment diagram for the ITTRFLIN register



The following table shows the ITTRFLIN register bit descriptions.

Table 9-659: ITTRFLIN bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
1	FLUSHIN	`0b0`	RO	In integration mode, this bit latches to 1 on a rising edge of the flushin input. It is cleared when this register is read, or when integration mode is disabled.
0	TRIGIN	`0b0`	RO	In integration mode, this bit latches to 1 on a rising edge of the trigin input. It is cleared when this register is read or when integration mode is disabled.

9.22.15 css600_tpiu Integration Test ATB Data Register 0, ITATBDATA0

This register indicates the value of the atdata_rx input in integration mode. Only 5 bits are readable through this register, the MSB of each of the four data bytes and the LSB.

Reads are allowed even in functional mode, but the register is disabled and does not get updated even if the inputs change. The reset value depends on the external source driving the inputs.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEEC`

Type

RO

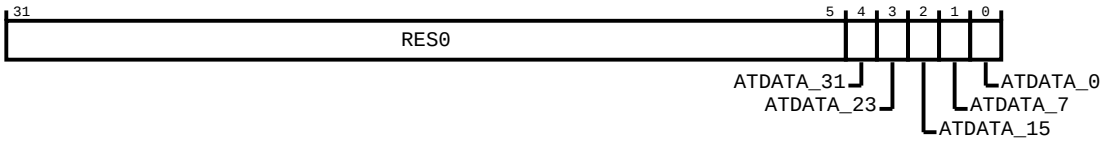
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITATBDATA0 register bit assignments.

Figure 9-639: Bit assignment diagram for the ITATBDATA0 register



The following table shows the ITATBDATA0 register bit descriptions.

Table 9-660: ITATBDATA0 bit descriptions

Bits	Name	Reset	Type	Description
31:5	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
4	ATDATA_31	0b0	RO	Reads the value of atdata_rx[31] during integration mode.
3	ATDATA_23	0b0	RO	Reads the value of atdata_rx[23] during integration mode.
2	ATDATA_15	0b0	RO	Reads the value of atdata_rx[15] during integration mode.
1	ATDATA_7	0b0	RO	Reads the value of atdata_rx[7] during integration mode.
0	ATDATA_0	0b0	RO	Reads the value of atdata_rx[0] during integration mode.

9.22.16 css600_tpiu Integration Test ATB Control Register 2, ITATBCTR2

This register enables control of the atready_rx, afvalid_rx, and syncreq_rx outputs in integration mode. Writes to this register are allowed in integration mode as well as functional mode. However, the programmed value is driven to the outputs only in integration mode.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEF0`

Type

WO

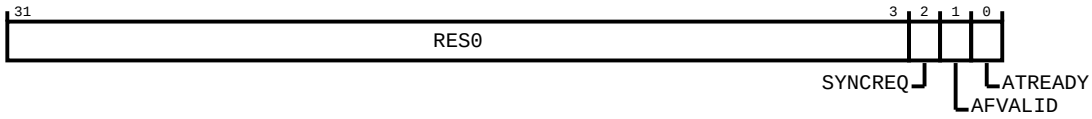
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITATBCTR2 register bit assignments.

Figure 9-640: Bit assignment diagram for the ITATBCTR2 register



The following table shows the ITATBCTR2 register bit descriptions.

Table 9-661: ITATBCTR2 bit descriptions

Bits	Name	Reset	Type	Description
31:3	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
2	SYNCREQ	0b0	WO	Sets the value of syncreq_rx in integration mode.
1	AFVALID	0b0	WO	Sets the value of afvalid_rx in integration mode.
0	ATREADY	0b0	WO	Sets the value of atready_rx in integration mode.

9.22.17 css600_tpiu Integration Test ATB Control Register 1, ITATBCTR1

This register indicates the value of the atid_rx input in integration mode. Reads are allowed even in functional mode, but the register is disabled and does not get updated even if the inputs change. The reset value depends on external source driving these inputs.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEF4`

Type

RO

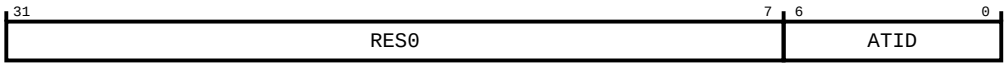
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITATBCTR1 register bit assignments.

Figure 9-641: Bit assignment diagram for the ITATBCTR1 register



The following table shows the ITATBCTR1 register bit descriptions.

Table 9-662: ITATBCTR1 bit descriptions

Bits	Name	Reset	Type	Description
31:7	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
6:0	ATID	0x0	RO	Reads the value of atid_rx[6:0] in integration mode.

9.22.18 css600_tpiu Integration Test ATB Control Register 0, ITATBCTR0

This register indicates the values of atvalid_rx, afready_rx, and atbytes_rx inputs in integration mode. Reads are allowed even in functional mode, but the register is disabled and does not get updated even if the inputs change. The reset value depends on an external source driving these inputs.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEF8

Type

RO

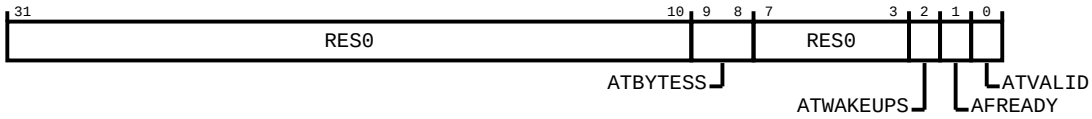
Reset value

0x00000000

Bit descriptions

The following figure shows the ITATBCTR0 register bit assignments.

Figure 9-642: Bit assignment diagram for the ITATBCTR0 register



The following table shows the ITATBCTR0 register bit descriptions.

Table 9-663: ITATBCTR0 bit descriptions

Bits	Name	Reset	Type	Description
31:10	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
9:8	ATBYTESS	0b00	RO	Reads the value of atbytes_rx[1:0] in integration mode.
7:3	**RES0**	0b00000	RO	Reserved bit or field with SBZP behavior.
2	ATWAKEUPS	0b0	RO	Reads the value of atwakeup_rx in integration mode.
1	AFREADY	0b0	RO	Reads the value of afready_rx in integration mode.
0	ATVALID	0b0	RO	Reads the value of atvalid_rx in integration mode.

9.22.19 css600_tpiu Integration Test Output Control Register, ITOUTCTR

This register enables control of the flushcomp output in integration mode. Writes to this register are allowed in integration mode, as well as functional mode. However, the programmed value is driven to the output pin only in integration mode.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xEFC

Type

WO

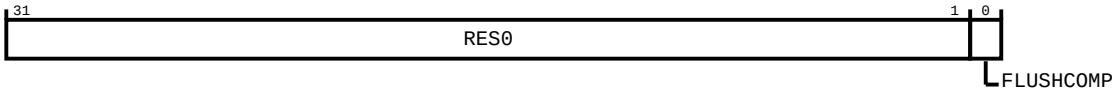
Reset value

0x00000000

Bit descriptions

The following figure shows the ITOUTCTR register bit assignments.

Figure 9-643: Bit assignment diagram for the ITOUTCTR register



The following table shows the ITOUTCTR register bit descriptions.

Table 9-664: ITOUTCTR bit descriptions

Bits	Name	Reset	Type	Description
31:1	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
0	FLUSHCOMP	0b0	WO	Sets the value of flushcomp in integration mode.

9.22.20 css600_tpiu Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

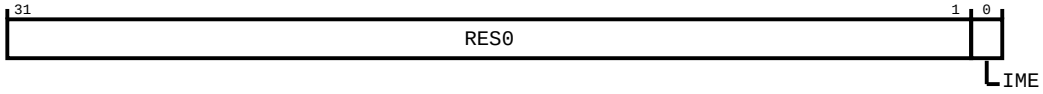
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-644: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-665: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.22.21 css600_tpiu Claim Tag Set Register, CLAIMSET

This register forms one half of the claim tag value. On writes, this location enables individual bits to be set. On reads, it returns the number of bits that can be set.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFA0

Type

RW

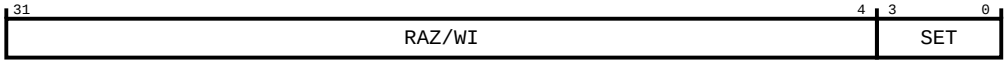
Reset value

0x0000000F

Bit descriptions

The following figure shows the CLAIMSET register bit assignments.

Figure 9-645: Bit assignment diagram for the CLAIMSET register



The following table shows the CLAIMSET register bit descriptions.

Table 9-666: CLAIMSET bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	`0x0`	RO	**RAZ/WI**.
3:0	SET	`0b1111`	RW	A bit-programmable register bank that sets the claim tag value. A read returns a logic 1 for all implemented locations.

9.22.22 css600_tpiu Claim Tag Clear Register, CLAIMCLR

This register forms one half of the claim tag value. On writes, this location enables individual bits to be cleared. On reads, it returns the current claim tag value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFA4`

Type

RW

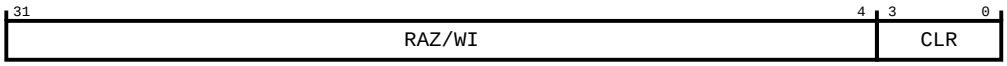
Reset value

`0x00000000`

Bit descriptions

The following figure shows the CLAIMCLR register bit assignments.

Figure 9-646: Bit assignment diagram for the CLAIMCLR register



The following table shows the CLAIMCLR register bit descriptions.

Table 9-667: CLAIMCLR bit descriptions

Bits	Name	Reset	Type	Description
31:4	**RAZ/WI**	0x0	RO	**RAZ/WI.**
3:0	CLR	0b0000	RW	A bit-programmable register bank that clears the claim tag value. It is zero at reset. It is used by software agents to signal to each other ownership of the hardware. It has no direct effect on the hardware itself.

9.22.23 css600_tpiu Authentication Status Register, AUTHSTATUS

Reports the current status of the authentication control signals.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFB8

Type

RO

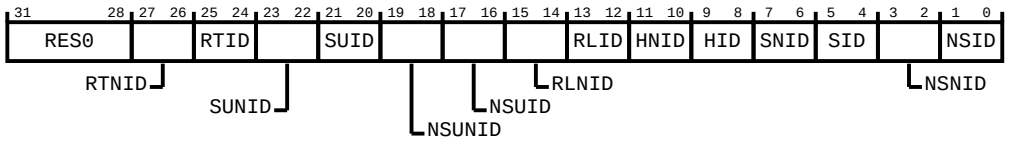
Reset value

0x00000000

Bit descriptions

The following figure shows the AUTHSTATUS register bit assignments.

Figure 9-647: Bit assignment diagram for the AUTHSTATUS register



The following table shows the AUTHSTATUS register bit descriptions.

Table 9-668: AUTHSTATUS bit descriptions

Bits	Name	Reset	Type	Description
31:28	**RES0**	0b0000	RO	Reserved bit or field with SBZP behavior.
27:26	RTNID	0b00	RO	Root non-invasive debug. **0b00** Separate Root non-invasive debug enable not implemented or Root state non-invasive debug features not implemented.

Bits	Name	Reset	Type	Description
25:24	RTID	0b00	RO	Root invasive debug. **0b00** Separate Root invasive debug enable not implemented or Root state invasive debug features not implemented.
23:22	SUNID	0b00	RO	Secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
21:20	SUID	0b00	RO	Secure unprivileged invasive debug. **0b00** Debug level is not supported.
19:18	NSUNID	0b00	RO	Non-secure unprivileged non-invasive debug. **0b00** Debug level is not supported.
17:16	NSUID	0b00	RO	Non-secure unprivileged invasive debug. **0b00** Debug level is not supported.
15:14	RLNID	0b00	RO	Realm non-invasive debug. **0b00** Separate Realm non-invasive debug enable not implemented or Realm state non-invasive debug features not implemented.
13:12	RLID	0b00	RO	Realm invasive debug. **0b00** Separate Realm invasive debug enable not implemented or Realm state invasive debug features not implemented.
11:10	HNID	0b00	RO	Hypervisor non-invasive debug. **0b00** Debug level is not supported.
9:8	HID	0b00	RO	Hypervisor invasive debug. **0b00** Debug level is not supported.
7:6	SNID	0b00	RO	Secure non-invasive debug. **0b00** Debug level is not supported.
5:4	SID	0b00	RO	Secure invasive debug. **0b00** Debug level is not supported.
3:2	NSNID	0b00	RO	Non-secure non-invasive debug. **0b00** Debug level is not supported.
1:0	NSID	0b00	RO	Non-secure invasive debug. **0b00** Debug level is not supported.

9.22.24 css600_tpiu Device Architecture Register, DEVARCH

Identifies the architect and architecture of a CoreSight component. The architect might differ from the designer of a component, for example Arm defines the architecture but another company designs and implements the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFBC`

Type

RO

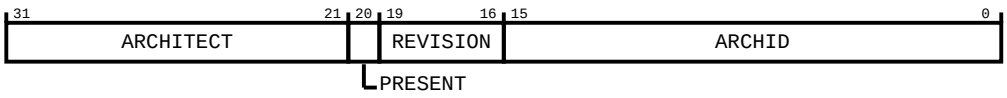
Reset value

`0x00000000`

Bit descriptions

The following figure shows the DEVARCH register bit assignments.

Figure 9-648: Bit assignment diagram for the DEVARCH register



The following table shows the DEVARCH register bit descriptions.

Table 9-669: DEVARCH bit descriptions

Bits	Name	Reset	Type	Description
31:21	ARCHITECT	`0x000`	RO	Defines the architect of the component
20	PRESENT	`0b0`	RO	Indicates the presence of this register **`0b0`** DEVARCH is not present
19:16	REVISION	`0b0000`	RO	Architecture revision. Returns the revision of the architecture that the ARCHID field specifies
15:0	ARCHID	`0x0000`	RO	Architecture ID. Returns a value that identifies the architecture of the component.

9.22.25 css600_tpiu Device Configuration Register, DEVID

This register is **IMPLEMENTATION DEFINED** for each Part Number and Designer. The register indicates the capabilities of the component.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFC8`

Type

RO

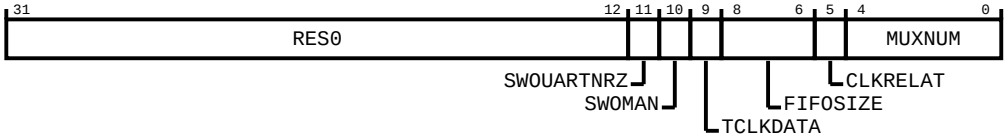
Reset value

`0x00000020`

Bit descriptions

The following figure shows the DEVID register bit assignments.

Figure 9-649: Bit assignment diagram for the DEVID register



The following table shows the DEVID register bit descriptions.

Table 9-670: DEVID bit descriptions

Bits	Name	Reset	Type	Description
31:12	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
11	SWOUARTNRZ	`0b0`	RO	Serial Wire Output, UART or NRZ support. Reads `0x0`, which indicates that Serial Wire Output, UART or NRZ, is not supported.
10	SWOMAN	`0b0`	RO	Serial Wire Output, Manchester-encoded format support. Reads `0x0`, which indicates that Serial Wire Output, Manchester-encoded format, is not supported.
9	TCLKDATA	`0b0`	RO	Trace Clock Plus Data support. Reads `0x0`, which indicates that trace clock and data is supported.
8:6	FIFOSIZE	`0b000`	RO	FIFO size in powers of 2. Reads `0x0`, indicating that the FIFO size is implementation-defined and is not visible in the programmers model.
5	CLKRELAT	`0b1`	RO	Relationship between clk and traceclkin. Reads `0x1` which indicates that these two clocks are asynchronous.
4:0	MUXNUM	`0b00000`	RO	Indicates a hidden level of input multiplexing. When non-zero, this value indicates the type of multiplexing on the input to the ATB. Currently only `0x00` is supported, that is, no multiplexing is present. This value helps detect the ATB structure.

9.22.26 css600_tpiu Device Type Identifier Register, DEVTYPE

A debugger can use this register to get information about a component that has an unrecognized Part number.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFCC`

Type

RO

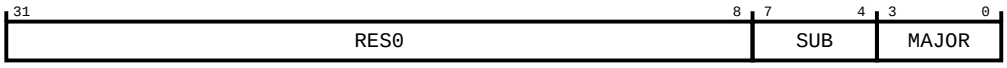
Reset value

`0x00000011`

Bit descriptions

The following figure shows the DEVTYPE register bit assignments.

Figure 9-650: Bit assignment diagram for the DEVTYPE register



The following table shows the DEVTYPE register bit descriptions.

Table 9-671: DEVTYPE bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	SUB	`0b0001`	RO	Minor classification. Returns `0x1`, indicating this component is a Trace Port.
3:0	MAJOR	`0b0001`	RO	Major classification. Returns `0x1`, indicating this component is a Trace Sink.

9.22.27 css600_tpiu Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

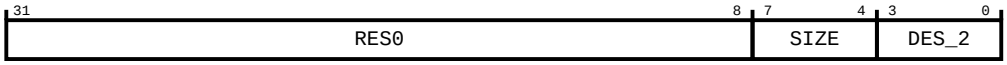
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-651: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-672: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.22.28 css600_tpiu Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

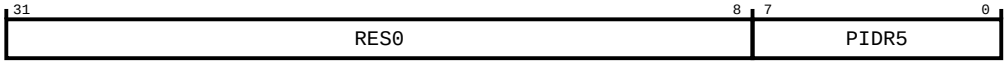
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-652: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-673: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.22.29 css600_tpiu Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD8

Type

RO

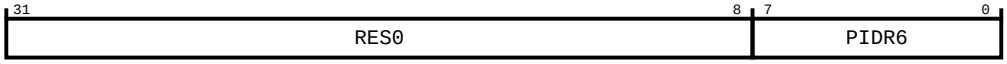
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-653: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-674: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.22.30 css600_tpiu Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

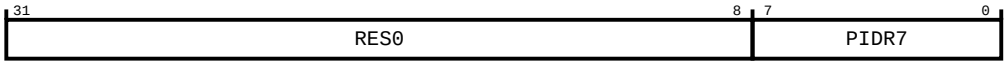
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-654: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-675: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.22.31 css600_tpiu Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

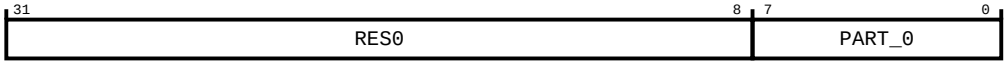
Reset value

0x000000E7

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-655: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-676: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0xE7	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.22.32 css600_tpiu Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

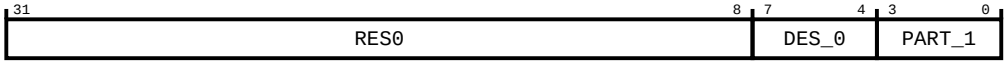
Reset value

0x000000B9

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-656: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-677: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b1001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.22.33 css600_tpiu Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

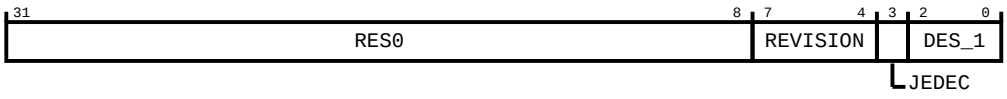
Reset value

0x0000004B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-657: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-678: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b0100	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.22.34 css600_tpiu Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFEC

Type

RO

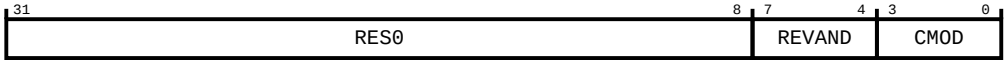
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-658: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-679: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	REVAND	0b0000	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is 0x0.
3:0	CMOD	0b0000	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is 0x0.

9.22.35 css600_tpiu Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

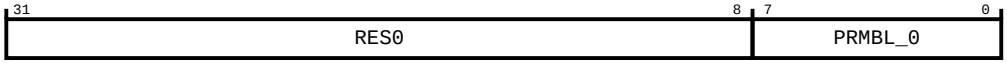
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-659: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-680: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.22.36 css600_tpiu Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF4`

Type

RO

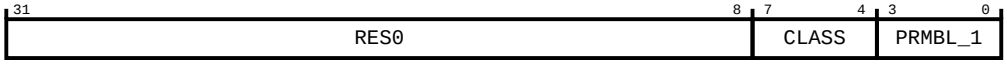
Reset value

`0x00000090`

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-660: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-681: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	`0b1001`	RO	Component class **`0b1001`** CoreSight component
3:0	PRMBL_1	`0b0000`	RO	Preamble. Returns `0x0`.

9.22.37 css600_tpiu Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

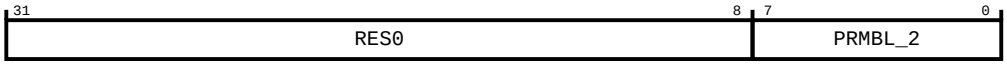
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-661: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-682: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.22.38 css600_tpiu Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

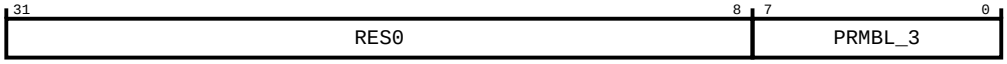
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-662: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-683: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.23 css600_tsgen introduction

The SoC-600 `css600_tsgen` component has two APB5 completer interfaces, and each interface has its own set of registers.

The `css600_tsgen` register interfaces are:

APB5_Completer_0

A control interface for programming the timestamp generator. See css600_tsgen register summary for APB5_Completer_0.

APB5_Completer_1

A read-only interface for reading the counter value. See css600_tsgen register summary for APB5_Completer_1.

9.24 css600_tsgen register summary for APB5_Completer_0

The APB5_Completer_0 interface in the SoC-600 `css600_tsgen` component is a control interface for programming the timestamp generator.

The summary of the registers is in order of address offset, and contains a description of the bitfields for each register.

Summary table

Table 9-684: css600_tsgen_APB5_Completer_0 register summary

Offset	Name	Type	Reset	Width	Description
0x000	CNTCR	RW	0x00000000	32-bit	Counter Control Register
0x004	CNTSR	RO	0x00000000	32-bit	Counter Status Register
0x008	CNTCVL	RW	0x00000000	32-bit	Current value of Counter[31:0]

Offset	Name	Type	Reset	Width	Description
0x00c	CNTCVU	RW	0x00000000	32-bit	Current value of Counter[63:32]
0x020	CNTFID0	RW	0x00000000	32-bit	Base Frequency ID register
0xef8	ITSTAT	RO	0x00000000	32-bit	Integration Test Status Register
0xf00	ITCTRL	RW	0x00000000	32-bit	Integration Mode Control Register
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x00000093	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B1	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000000B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x000000F0	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.24.1 css600_tsgen Counter Control Register, CNTCR

The CNTCR register controls the counter increments.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x000

Type

RW

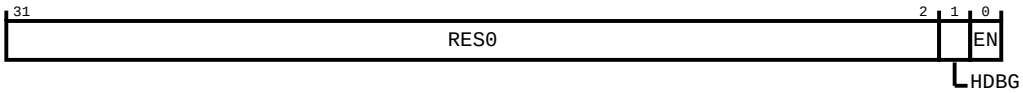
Reset value

0x00000000

Bit descriptions

The following figure shows the CNTCR register bit assignments.

Figure 9-663: Bit assignment diagram for the CNTCR register



The following table shows the CNTCR register bit descriptions.

Table 9-685: CNTCR bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	HDBG	0b0	RW	Halt On Debug: **0b0** Do not halt on debug. The halt_req signal into the counter has no effect. **0b1** Halt on debug. When the halt_req pulse is received, the count value is held static.
0	EN	0b0	RW	Enable Bit. **0b0** The counter is disabled. Count is not incrementing. **0b1** The counter is enabled. Count is incrementing.

9.24.2 css600_tsgen Counter Status Register, CNTSR

The CNTSR register identifies the status of the counter.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x004

Type

RO

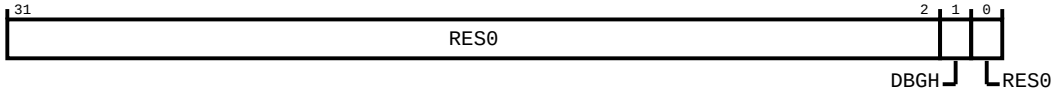
Reset value

0x00000000

Bit descriptions

The following figure shows the CNTSR register bit assignments.

Figure 9-664: Bit assignment diagram for the CNTSR register



The following table shows the CNTSR register bit descriptions.

Table 9-686: CNTSR bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	DBGH	0b0	RO	Indicates whether the counter is halted because the Halt-on-debug signal is asserted: **0b0** Counter is not halted **0b1** Counter is halted
0	**RES0**	0b0	RO	Reserved bit or field with SBZP behavior.

9.24.3 css600_tsgen Current value of Counter[31:0], CNTCVL

The CNTCVL register reads or writes the lower 32 bits of the current counter value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x008

Type

RW

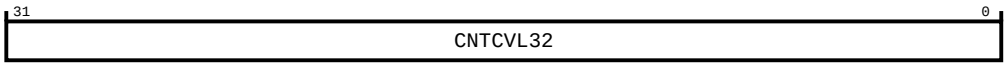
Reset value

0x00000000

Bit descriptions

The following figure shows the CNTCVL register bit assignments.

Figure 9-665: Bit assignment diagram for the CNTCVL register



The following table shows the CNTCVL register bit descriptions.

Table 9-687: CNTCVL bit descriptions

Bits	Name	Reset	Type	Description
31:0	CNTCVL32	0x0	RW	Reads to this register return the lower 32 bits of the current timestamp counter value. To change the current timestamp value, write the lower 32 bits of the new value to this register before writing the upper 32 bits to CNTCVU. The timestamp value is not changed until the CNTCVU register is written to.

9.24.4 css600_tsgen Current value of Counter[63:32], CNTCVU

The CNTCVU register reads or writes the upper 32 bits of the current counter value. The control interface must clear the CNTCR.EN bit or set CNTCR.HDBG and hlt_dbg asserted on the input to stop the counter, before writing to this register.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x00C

Type

RW

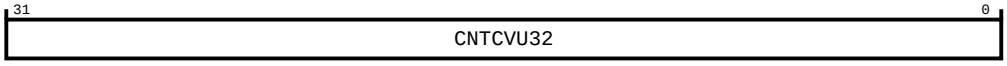
Reset value

0x00000000

Bit descriptions

The following figure shows the CNTCVU register bit assignments.

Figure 9-666: Bit assignment diagram for the CNTCVU register



The following table shows the CNTCVU register bit descriptions.

Table 9-688: CNTCVU bit descriptions

Bits	Name	Reset	Type	Description
31:0	CNTCVU32	0x0	RW	Reads to this register return the upper 32 bits of the current timestamp counter value. To change the current timestamp value, write the lower 32 bits of the new value to CNTCVL before writing the upper 32 bits to this register. The 64-bit timestamp value is updated with the value from both writes when this register is written to.

9.24.5 css600_tsgen Base Frequency ID register, CNTFID0

You must program the CNTFID0 register to match the clock frequency of the timestamp generator, in ticks per second. For example, for a 50 MHz clock, program 0x02FAF080. The real-time speed of the counter does not depend on the value of this register.

This register reports, to the reader, the speed of the counter as programmed by the system firmware.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0x020`

Type

RW

Reset value

`0x00000000`

Bit descriptions

The following figure shows the CNTFID0 register bit assignments.

Figure 9-667: Bit assignment diagram for the CNTFID0 register

The following table shows the CNTFID0 register bit descriptions.

Table 9-689: CNTFID0 bit descriptions

Bits	Name	Reset	Type	Description
31:0	Freq	`0x0`	RW	Frequency in number of ticks per second. Up to 4GHz can be specified.

9.24.6 css600_tsgen Integration Test Status Register, ITSTAT

The ITSTAT register views the halt_req and restart_req values.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xEF8`

Type

RO

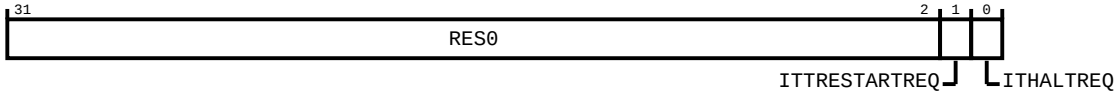
Reset value

`0x00000000`

Bit descriptions

The following figure shows the ITSTAT register bit assignments.

Figure 9-668: Bit assignment diagram for the ITSTAT register



The following table shows the ITSTAT register bit descriptions.

Table 9-690: ITSTAT bit descriptions

Bits	Name	Reset	Type	Description
31:2	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
1	ITTRESTARTREQ	0b0	RO	Integration Test Restart Request status of the restart_req input. Integration testing mode: Behaves as a sticky bit and latches to 1 when tsgen receives restart request. Cleared on reading this register. If restart_req is asserted in the same cycle as an APB read of this register, the read takes priority and the register is cleared as a result. Always returns 0 in normal functional mode.
0	ITHALTREQ	0b0	RO	Integration Test Halt Request status of the halt_req input. Integration testing mode: Behaves as a sticky bit and latches to 1 when tsgen receives halt request. Cleared on reading this register. If halt_req is asserted in the same cycle as an APB read of this register, the read takes priority and the register is cleared as a result. Always returns 0 in normal functional mode.

9.24.7 css600_tsgen Integration Mode Control Register, ITCTRL

The Integration Mode Control register is used to switch between functional mode and integration mode.

After switching to integration mode and performing integration tests or topology detection, you must reset the system to ensure correct behavior of CoreSight and other connected system components.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xF00

Type

RW

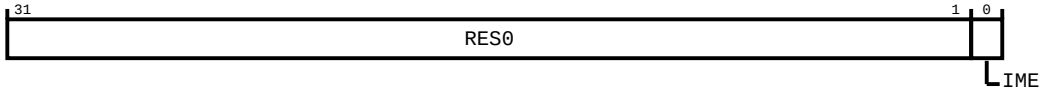
Reset value

0x00000000

Bit descriptions

The following figure shows the ITCTRL register bit assignments.

Figure 9-669: Bit assignment diagram for the ITCTRL register



The following table shows the ITCTRL register bit descriptions.

Table 9-691: ITCTRL bit descriptions

Bits	Name	Reset	Type	Description
31:1	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
0	IME	0b0	RW	Integration Mode Enable. **0b0** The component must enter functional mode. **0b1** The component must enter integration mode, and enable support for topology detection and integration testing.

9.24.8 css600_tsgen Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

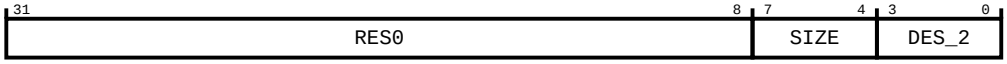
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-670: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-692: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.24.9 css600_tsgen Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD4

Type

RO

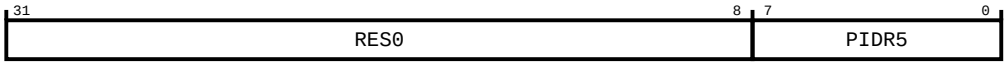
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-671: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-693: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.24.10 css600_tsgen Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD8

Type

RO

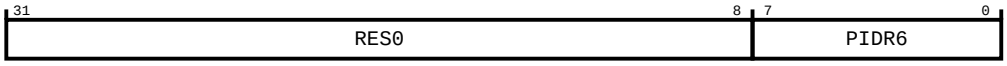
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-672: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-694: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.24.11 css600_tsgen Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

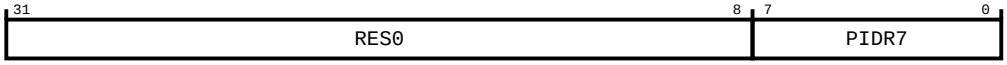
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-673: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-695: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.24.12 css600_tsgen Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

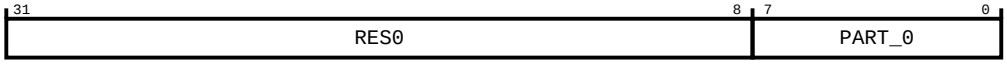
Reset value

0x00000093

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-674: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-696: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PART_0	0x93	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.24.13 css600_tsgen Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

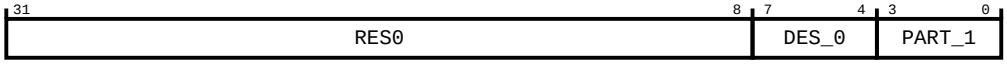
Reset value

0x000000B1

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-675: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-697: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b0001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.24.14 css600_tsgen Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE8

Type

RO

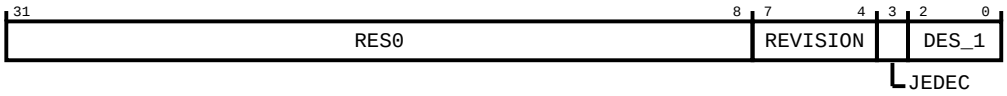
Reset value

0x0000000B

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-676: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-698: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	0b0000	RO	Revision. It is an incremental value starting at 0x0 for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	0b1	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	0b011	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.24.15 css600_tsgen Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

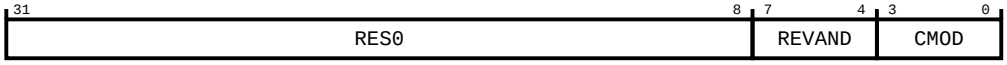
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-677: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-699: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	`0b0000`	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is `0x0`.
3:0	CMOD	`0b0000`	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is `0x0`.

9.24.16 css600_tsgen Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF0

Type

RO

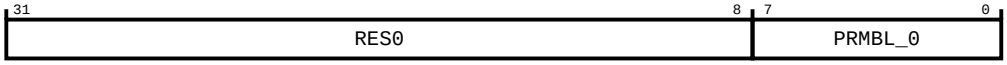
Reset value

0x0000000D

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-678: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-700: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.24.17 css600_tsgen Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF4

Type

RO

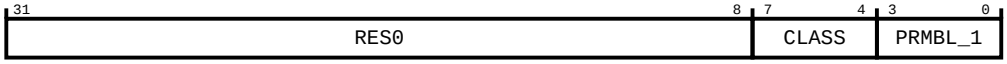
Reset value

0x000000F0

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-679: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-701: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	`0b1111`	RO	Component class **`0b1111`** CoreLink, PrimeCell, or system component
3:0	PRMBL_1	`0b0000`	RO	Preamble. Returns `0x0`.

9.24.18 css600_tsgen Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF8`

Type

RO

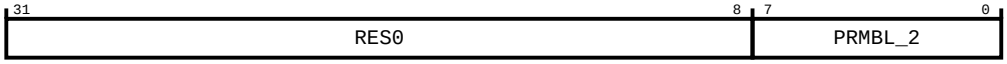
Reset value

`0x00000005`

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-680: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-702: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.24.19 css600_tsgen Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFFC

Type

RO

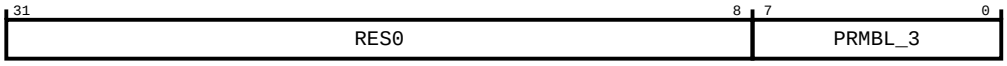
Reset value

0x000000B1

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-681: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-703: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	0xB1	RO	Preamble. Returns 0xB1.

9.25 css600_tsgen register summary for APB5_Completer_1

This section describes the css600_tsgen_registers registers. It contains a summary of the registers, in order of address offset, and a description of the bitfields for each register.

Summary table

Table 9-704: css600_tsgen_APB5_Completer_1 register summary

Offset	Name	Type	Reset	Width	Description
0x000	CNTCVLREAD	RO	0x00000000	32-bit	Current value of Counter[31:0]
0x004	CNTCVUREAD	RO	0x00000000	32-bit	Current value of Counter[63:32]
0xfd0	PIDR4	RO	0x00000004	32-bit	Peripheral Identification Register 4
0xfd4	PIDR5	RO	0x00000000	32-bit	Peripheral Identification Register 5
0xfd8	PIDR6	RO	0x00000000	32-bit	Peripheral Identification Register 6
0xfdc	PIDR7	RO	0x00000000	32-bit	Peripheral Identification Register 7
0xfe0	PIDR0	RO	0x00000093	32-bit	Peripheral Identification Register 0
0xfe4	PIDR1	RO	0x000000B1	32-bit	Peripheral Identification Register 1
0xfe8	PIDR2	RO	0x0000000B	32-bit	Peripheral Identification Register 2
0xfec	PIDR3	RO	0x00000000	32-bit	Peripheral Identification Register 3
0xff0	CIDR0	RO	0x0000000D	32-bit	Component Identification Register 0
0xff4	CIDR1	RO	0x000000F0	32-bit	Component Identification Register 1
0xff8	CIDR2	RO	0x00000005	32-bit	Component Identification Register 2
0xffc	CIDR3	RO	0x000000B1	32-bit	Component Identification Register 3

9.25.1 css600_tsgen Current value of Counter[31:0], CNTCVLREAD

The CNTCVLREAD register reads the lower 32 bits of the current counter value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x000

Type

RO

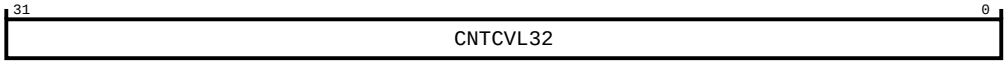
Reset value

0x00000000

Bit descriptions

The following figure shows the CNTCVLREAD register bit assignments.

Figure 9-682: Bit assignment diagram for the CNTCVLREAD register



The following table shows the CNTCVLREAD register bit descriptions.

Table 9-705: CNTCVLREAD bit descriptions

Bits	Name	Reset	Type	Description
31:0	CNTCVL32	0x0	RO	The lower 32 bits of the current timestamp counter value

9.25.2 css600_tsgen Current value of Counter[63:32], CNTCVUREAD

The CNTCVUREAD register reads the upper 32 bits of the current counter value.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0x004

Type

RO

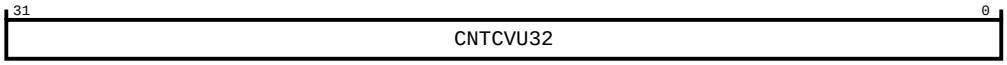
Reset value

0x00000000

Bit descriptions

The following figure shows the CNTCVUREAD register bit assignments.

Figure 9-683: Bit assignment diagram for the CNTCVUREAD register



The following table shows the CNTCVUREAD register bit descriptions.

Table 9-706: CNTCVUREAD bit descriptions

Bits	Name	Reset	Type	Description
31:0	CNTCVU32	0x0	RO	The upper 32 bits of the current timestamp counter value

9.25.3 css600_tsgen Peripheral Identification Register 4, PIDR4

The PIDR4 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFD0

Type

RO

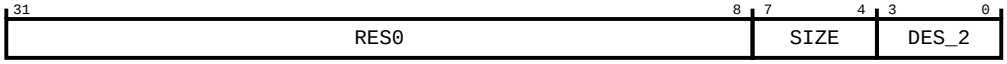
Reset value

0x00000004

Bit descriptions

The following figure shows the PIDR4 register bit assignments.

Figure 9-684: Bit assignment diagram for the PIDR4 register



The following table shows the PIDR4 register bit descriptions.

Table 9-707: PIDR4 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	SIZE	0b0000	RO	Indicates the memory size that is used by this component. Returns 0 indicating that the component uses an **UNKNOWN** number of 4KB blocks. Using the SIZE field to indicate the size of the component is deprecated.
3:0	DES_2	0b0100	RO	JEP106 continuation code. Together with PIDR2.DES_1 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.25.4 css600_tsgen Peripheral Identification Register 5, PIDR5

The PIDR5 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD4`

Type

RO

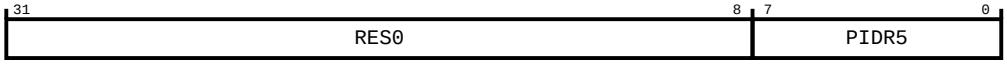
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR5 register bit assignments.

Figure 9-685: Bit assignment diagram for the PIDR5 register



The following table shows the PIDR5 register bit descriptions.

Table 9-708: PIDR5 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR5	0x0	RO	Reserved.

9.25.5 css600_tsgen Peripheral Identification Register 6, PIDR6

The PIDR6 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFD8`

Type

RO

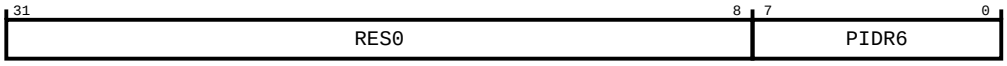
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR6 register bit assignments.

Figure 9-686: Bit assignment diagram for the PIDR6 register



The following table shows the PIDR6 register bit descriptions.

Table 9-709: PIDR6 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR6	0x0	RO	Reserved.

9.25.6 css600_tsgen Peripheral Identification Register 7, PIDR7

The PIDR7 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFDC

Type

RO

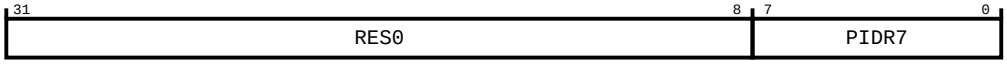
Reset value

0x00000000

Bit descriptions

The following figure shows the PIDR7 register bit assignments.

Figure 9-687: Bit assignment diagram for the PIDR7 register



The following table shows the PIDR7 register bit descriptions.

Table 9-710: PIDR7 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PIDR7	0x0	RO	Reserved.

9.25.7 css600_tsgen Peripheral Identification Register 0, PIDR0

The PIDR0 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE0

Type

RO

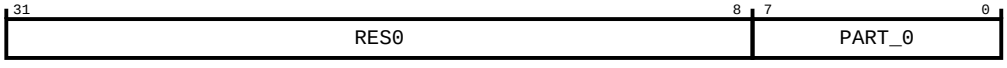
Reset value

0x00000093

Bit descriptions

The following figure shows the PIDR0 register bit assignments.

Figure 9-688: Bit assignment diagram for the PIDR0 register



The following table shows the PIDR0 register bit descriptions.

Table 9-711: PIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.

Bits	Name	Reset	Type	Description
7:0	PART_0	0x93	RO	Part number, bits[7:0]. Taken together with PIDR1.PART_1 it indicates the component. The Part Number is selected by the designer of the component.

9.25.8 css600_tsgen Peripheral Identification Register 1, PIDR1

The PIDR1 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFE4

Type

RO

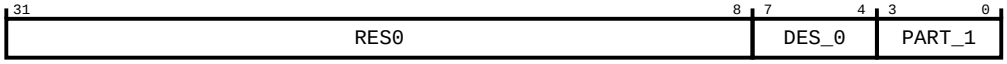
Reset value

0x000000B1

Bit descriptions

The following figure shows the PIDR1 register bit assignments.

Figure 9-689: Bit assignment diagram for the PIDR1 register



The following table shows the PIDR1 register bit descriptions.

Table 9-712: PIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	DES_0	0b1011	RO	JEP106 identification code, bits[3:0]. Together with PIDR4.DES_2 and PIDR2.DES_1, they indicate the designer of the component and not the implementer, except where the two are the same.
3:0	PART_1	0b0001	RO	Part number, bits[11:8]. Taken together with PIDR0.PART_0 it indicates the component. The Part Number is selected by the designer of the component.

9.25.9 css600_tsgen Peripheral Identification Register 2, PIDR2

The PIDR2 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFE8`

Type

RO

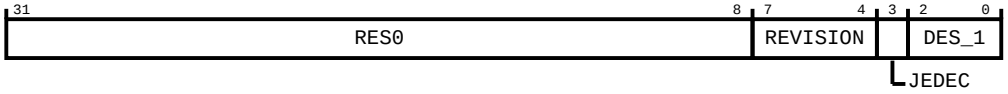
Reset value

`0x0000000B`

Bit descriptions

The following figure shows the PIDR2 register bit assignments.

Figure 9-690: Bit assignment diagram for the PIDR2 register



The following table shows the PIDR2 register bit descriptions.

Table 9-713: PIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVISION	`0b0000`	RO	Revision. It is an incremental value starting at `0x0` for the first design of a component. See the Component list in Chapter 1 for information on the RTL revision of the component.
3	JEDEC	`0b1`	RO	1 - Always set. Indicates that a JEDEC assigned value is used.
2:0	DES_1	`0b011`	RO	JEP106 identification code, bits[6:4]. Together, with PIDR4.DES_2 and PIDR1.DES_0, they indicate the designer of the component and not the implementer, except where the two are the same.

9.25.10 css600_tsgen Peripheral Identification Register 3, PIDR3

The PIDR3 register is part of the set of peripheral identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFEC`

Type

RO

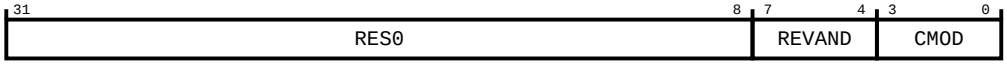
Reset value

`0x00000000`

Bit descriptions

The following figure shows the PIDR3 register bit assignments.

Figure 9-691: Bit assignment diagram for the PIDR3 register



The following table shows the PIDR3 register bit descriptions.

Table 9-714: PIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:4	REVAND	`0b0000`	RO	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is `0x0`.
3:0	CMOD	`0b0000`	RO	Customer Modified. Where the component is reusable IP this value indicates if the customer has modified the behavior of the component. In most cases this field is `0x0`.

9.25.11 css600_tsgen Component Identification Register 0, CIDR0

The CIDR0 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF0`

Type

RO

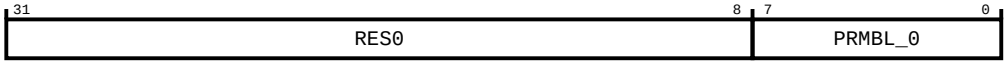
Reset value

`0x0000000D`

Bit descriptions

The following figure shows the CIDR0 register bit assignments.

Figure 9-692: Bit assignment diagram for the CIDR0 register



The following table shows the CIDR0 register bit descriptions.

Table 9-715: CIDR0 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_0	0xD	RO	Preamble. Returns 0x0D.

9.25.12 css600_tsgen Component Identification Register 1, CIDR1

The CIDR1 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFF4`

Type

RO

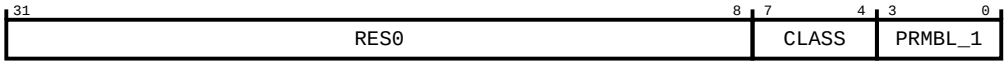
Reset value

`0x000000F0`

Bit descriptions

The following figure shows the CIDR1 register bit assignments.

Figure 9-693: Bit assignment diagram for the CIDR1 register



The following table shows the CIDR1 register bit descriptions.

Table 9-716: CIDR1 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:4	CLASS	0b1111	RO	Component class **0b1111** CoreLink, PrimeCell, or system component
3:0	PRMBL_1	0b0000	RO	Preamble. Returns 0x0.

9.25.13 css600_tsgen Component Identification Register 2, CIDR2

The CIDR2 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

0xFF8

Type

RO

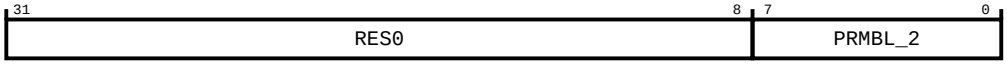
Reset value

0x00000005

Bit descriptions

The following figure shows the CIDR2 register bit assignments.

Figure 9-694: Bit assignment diagram for the CIDR2 register



The following table shows the CIDR2 register bit descriptions.

Table 9-717: CIDR2 bit descriptions

Bits	Name	Reset	Type	Description
31:8	**RES0**	0x0	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_2	0x5	RO	Preamble. Returns 0x05.

9.25.14 css600_tsgen Component Identification Register 3, CIDR3

The CIDR3 register is part of the set of component identification registers.

Attributes

Its characteristics are:

Width

32-bit

Address offset

`0xFFC`

Type

RO

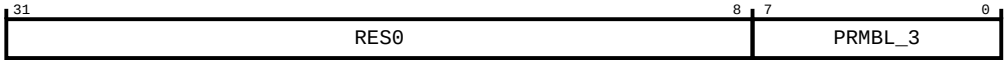
Reset value

`0x000000B1`

Bit descriptions

The following figure shows the CIDR3 register bit assignments.

Figure 9-695: Bit assignment diagram for the CIDR3 register



The following table shows the CIDR3 register bit descriptions.

Table 9-718: CIDR3 bit descriptions

Bits	Name	Reset	Type	Description
31:8	RES0	`0x0`	RO	Reserved bit or field with SBZP behavior.
7:0	PRMBL_3	`0xB1`	RO	Preamble. Returns `0xB1`.

Proprietary Notice

This document is protected by copyright and other related rights and the use or implementation of the information contained in this document may be protected by one or more patents or pending patent applications. No part of this document may be reproduced in any form by any means without the express prior written permission of Arm Limited ("Arm"). No license, express or implied, by estoppel or otherwise to any intellectual property rights is granted by this document unless specifically stated.

Your access to the information in this document is conditional upon your acceptance that you will not use or permit others to use the information for the purposes of determining whether the subject matter of this document infringes any third party patents.

The content of this document is informational only. Any solutions presented herein are subject to changing conditions, information, scope, and data. This document was produced using reasonable efforts based on information available as of the date of issue of this document. The scope of information in this document may exceed that which Arm is required to provide, and such additional information is merely intended to further assist the recipient and does not represent Arm's view of the scope of its obligations. You acknowledge and agree that you possess the necessary expertise in system security and functional safety and that you shall be solely responsible for compliance with all legal, regulatory, safety and security related requirements concerning your products, notwithstanding any information or support that may be provided by Arm herein. In addition, you are responsible for any applications which are used in conjunction with any Arm technology described in this document, and to minimize risks, adequate design and operating safeguards should be provided for by you.

This document may include technical inaccuracies or typographical errors. THIS DOCUMENT IS PROVIDED "AS IS". ARM PROVIDES NO REPRESENTATIONS AND NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NON-INFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO THE DOCUMENT. For the avoidance of doubt, Arm makes no representation with respect to, and has undertaken no analysis to identify or understand the scope and content of, any patents, copyrights, trade secrets, trademarks, or other rights.

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL ARM BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Reference by Arm to any third party's products or services within this document is not an express or implied approval or endorsement of the use thereof.

This document consists solely of commercial items. You shall be responsible for ensuring that any permitted use, duplication, or disclosure of this document complies fully with any relevant

export laws and regulations to assure that this document or any portion thereof is not exported, directly or indirectly, in violation of such export laws. Use of the word “partner” in reference to Arm’s customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of this document shall prevail.

The validity, construction and performance of this notice shall be governed by English Law.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

PRE-1121-V1.0

Product and document information

Read the information in these sections to understand the release status of the product and documentation, and the conventions used in Arm documents.

Product status

All products and services provided by Arm require deliverables to be prepared and made available at different levels of completeness. The information in this document indicates the appropriate level of completeness for the associated deliverables.

Product completeness status

The information in this document is Final, that is for a developed product.

Product revision status

The r7p1 identifier indicates the revision status of the product described in this manual, where:

`rx` Identifies the major revision of the product.
`py` Identifies the minor revision or modification status of the product.

Revision history

These sections can help you understand how the document has changed over time.

Document release information

The Document history table gives the issue number and the released date for each released issue of this document.

Document history

Issue	Date	Confidentiality	Change
0701-17	30 June 2025	Non-Confidential	First REL release for r7p1
0700-16	31 March 2025	Non-Confidential	First REL release for r7p0
0600-15	7 March 2024	Non-Confidential	First REL release for r6p0
0500-14	5 December 2023	Non-Confidential	Second REL release for r5p0
0500-13	30 July 2023	Non-Confidential	First REL release for r5p0
0402-00	6 May 2022	Non-Confidential	First REL release for r4p2

Issue	Date	Confidentiality	Change
0401-00	26 November 2020	Non-Confidential	First REL release for r4p1
0400-01	16 September 2020	Non-Confidential	Second REL release for r4p0
0400-00	16 May 2020	Non-Confidential	First REL release for r4p0
0302-00	6 December 2019	Non-Confidential	First early release for r3p2
0301-01	13 May 2019	Non-Confidential	Second early release for r3p1
0301-00	30 April 2019	Non-Confidential	First early access release for r3p1
0300-00	18 May 2018	Non-Confidential	First release for r3p0
0200-00	8 December 2017	Non-Confidential	First release for r2p0
0100-00	1 August 2017	Non-Confidential	First release for r1p0
0000-01	11 May 2017	Non-Confidential	Second release for r0p0
0000-00	1 March 2017	Non-Confidential	First release for r0p0

Change history

The Change history tables describe the technical changes between released issues of this document in reverse order. Issue numbers match the revision history in Document release information on page 833.

Table 2: Differences between issue 0700-16 and issue 0701-17

Change	Location
Added implementation-defined registers.	css600_apv1adapter register summary
Corrected description of DBGH field in register.	css600_tsgen Counter Status Register, CNTSR
Added note to `css600_tmc_etr` Embedded Trace Router (ETR) description.	Trace memory controller
Removed text describing TMC standard usage models from section.	Trace memory controller
Added text describing TMC standard usage models to subsections.	• Circular Buffer mode • Software FIFO mode 1 • Software FIFO mode 2 • Hardware FIFO mode
Corrected version of css600_cti component from r1p0_1 to r1p0_2 in SoC-600 component list table.	Component list

Table 3: Differences between issue 0600-15 and issue 0700-16

Change	Location
Text correction. Corrected “trace buffer” to “trace memory”.	• css600_tmc_etr Data Buffer Address Low Register, DBALO • css600_tmc_etr Data Buffer Address High Register, DBAHI
Small text changes to references to interrupt signal.	• CATU interrupt interface • Address validation • Error handling
Corrected TMC standard usage models SWF1 and SWF2.	• Software FIFO mode 1 • Software FIFO mode 2

Table 4: Differences between issue 0500-14 and issue 0600-15

Change	Location
Updated product revisions.	Product revisions
Updated the AXI protocol specification issue.	Useful resources on page 842
Updated component versions, revisions, and IP-XACT versions.	Component list
Added MECID support.	AXI-AP features
Corrected the MTE support description.	AXI transfers
In a table column header, replaced MTE_PRESENT with Component in use.	AXI access generation
Added new section for MECID.	Memory Encryption Contexts ID
Added extra content and reorganized into sub-sections.	Root and realm transactions
Added translation stash configuration signals to the `css600_tmc_etr`.	Trace Memory Controller
Added the AXI Control Register 1.	AXICTL1

Table 5: Differences between issue 0500-13 and issue 0500-14

Change	Location
Added the missing state section in the architectural state machine description.	Stopped
Corrected step 3 of the standard usage models for the Software FIFO mode 1 (SWF1) and Software FIFO mode 2 (SWF2) modes.	Software FIFO mode 1 and Software FIFO mode 2
Added the previously missing bit field information to some register descriptions and corrections to some bit field descriptions.	Programmers model
Added the register information for the APB5_Completer_1 interface.	css600_tsgen introduction

Table 6: Differences between issue 0402-00 and issue 0500-13

Change	Location
Updated product revisions.	Product revisions
Updated component versions and revisions.	Component list
Updated feature list.	Features
Terminology and naming updates.	Throughout
Added RME support and features to MEM-AP.	Memory access ports
Added RME support APB Access Port.	APB Access Port
Added RME support AHB Access Port.	AHB Access Port

Change	Location
Clarified information on burst type for AXI transfers.	AXI transfers
Added information on new RME support.	Root and realm transactions
Improved information on MBIST interface.	MBIST interface
ETB, ETF, ETR, and ETS now provided as components, not configurations.	Trace Memory Controller
Narrow timestamp components removed.	Timestamp components functional description
PID and revision changed for some PIL register reset values.	Processor Integration Layer components

Table 7: Differences between issue 0401-00 and issue 0402-00

Change	Location
Updated product revisions.	Product revisions
Corrected and updated component versions and revisions.	Component list
Updated TMC mode information.	• Trace Memory Controller • Architectural state machine • [CB mode] • Standard usage models
Included missing figures.	Example configuration scenarios
Clarified interface information.	Cortex-A9 PIL overview
• Added field type to assignments tables. • Updated some reset values. • Clarified Prot function. • Corrected some **RAZ/WI** fields to **RES0**.	Programmers model

Table 8: Differences between issue 0400-01 and issue 0401-00

Change	Location
Updated product revisions.	Product revisions
Updated component list for `css600_tmc`.	Component list
Corrected introduction to base registers.	• css600_ahbap register summary • css600_axiap register summary • css600_axiap_mte register summary
• Corrected register summary PIDR2 reset value. • Corrected REVISION reset values for ETB, ETF, ETR, and ETS registers.	• css600_tmc_etb register summary • css600_tmc_etf register summary • css600_tmc_etr register summary • css600_tmc_ets register summary
Corrected REVAND information.	• css600_apbrom_32bit register summary • css600_apbrom_64bit register summary • css600_apbrom_gpr_32bit register summary • css600_apbrom_gpr_64bit register summary

Table 9: Differences between issue 0400-00 and issue 0400-01

Change	Location
Corrected document titles and numbers.	Throughout the manual
Corrected information about TMC registers.	Reads from TMC registers
Updated CATU initializing information.	Initializing the CATU
Updated `css600_apbap` register bit assignments.	css600_apbap register summary
Updated `css600_ahbap` register bit assignments.	css600_ahbap register summary
Updated `css600_axiap` register bit assignments.	css600_axiap register summary
Updated `css600_apv1adapter` register bit assignments.	css600_apv1adapter register summary
Updated `css600_jtagap` register bit assignments.	css600_jtagap register summary
Updated `css600_apbrom` register bit assignments.	• css600_apbrom_32bit register summary • css600_apbrom_64bit register summary
Updated `css600_apbrom_gpr` register bit assignments.	• css600_apbrom_gpr_32bit register summary • css600_apbrom_gpr_64bit register summary
Updated `css600_atbfunnel_prog` register bit assignments.	css600_atbfunnel_prog register summary
Updated `css600_atbreplicator_prog` register bit assignments.	css600_atbreplicator_prog register summary
Updated `css600_tmc_etb` RSZ and DEVID reset values, and register bit assignments.	css600_tmc_etb register summary
Updated `css600_tmc_etf` RSZ and DEVID reset values, and register bit assignments.	css600_tmc_etf register summary
Updated `css600_tmc_etr` DEVID reset value, and register bit assignments.	css600_tmc_etr register summary
Updated `css600_tmc_ets` RSZ, RRD, and DEVID reset values, and register bit assignments.	css600_tmc_ets register summary
Updated `css600_tpiu` EXTCTLIN and PIDR2 reset values, and register bit assignments.	css600_tpiu register summary
Updated `css600_catu` DEVID reset value, and register bit assignments.	css600_catu register summary
Updated `css600_cti` DEVID and PIDR reset values, and register bit assignments.	css600_cti register summary

Table 10: Differences between issue 0302-00 and issue 0400-00

Change	Location
Updated AHB-AP component.	Product revisions
Updated TMC information.	AXI manager interface
Updated CATU information.	CATU AXI manager
Added AXIAP-MTE component.	css600_axiap_mte register summary

Table 11: Differences between issue 0301-01 and issue 0302-00

Change	Location
Updated product revisions.	Product revisions
Updated DP components.	css600_dp register summary

Table 12: Differences between issue 0300-00 and issue 0301-01

Change	Location
Updated product revisions.	Product revisions
Updated various components, in particular ATBFUNNEL, ATBREPLICATOR, TMC_ETB, TMC_ETF, TMC_ETR, TMC_ETS, and TPIU.	Component list

Table 13: Differences between issue 0200-00 and issue 0300-00

Change	Location
Updated product revisions.	Product revisions
Added TPIU component.	Component list and css600_tpiu register summary

Table 14: Differences between issue 0100-00 and issue 0200-00

Change	Location
Updated product revisions.	Product revisions
Added CATU component.	CoreSight Address Translation Unit and css600_catu register summary
Added PIL components.	Cortex-A5 PIL overview, Cortex-A8 PIL overview, Cortex-A9 PIL overview, Cortex-R4 PIL overview, Cortex-R5 PIL overview, Cortex-M0 PIL overview, Cortex-M3 PIL overview

Table 15: Differences between issue 0000-01 and issue 0100-00

Change	Location
Updated product revisions.	Product revisions
Added Narrow Timestamp components.	Timestamp components functional description
Added PIL component.	Processor Integration Layer components
Component list has been updated.	Component list

Table 16: Differences between issue 0000-00 and issue 0000-01

Change	Location
Updated component list.	Component list
Added DAP components chapter.	DAP components functional description
Added APB infrastructure components chapter.	APB infrastructure components functional description
Added ATB infrastructure components chapter.	AMBA Trace Bus infrastructure components functional description
Added Timestamp components chapter.	Timestamp components functional description
Added Embedded cross-trigger components chapter.	Embedded Cross Trigger components functional description

Change	Location
Added Authentication components chapter.	Authentication components functional description
Updated DP programmers model section.	css600_dp register summary

Table 17: Issue 0000-00

Change	Location
First release	-

Conventions

The following subsections describe conventions used in Arm documents.

Glossary

The Arm Glossary is a list of terms used in Arm documentation, together with definitions for those terms. The Arm Glossary does not contain terms that are industry standard unless the Arm meaning differs from the generally accepted meaning.

See the Arm Glossary for more information: developer.arm.com/glossary.

Typographic conventions

Arm documentation uses typographical conventions to convey specific meaning.

Convention	Use
italic	Citations.
bold	Terms in descriptive lists, where appropriate.
`monospace`	Text that you can enter at the keyboard, such as commands, file and program names, and source code.
`monospace underline`	A permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.
`<and>`	Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example: `MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2>`
SMALL CAPITALS	Terms that have specific technical meanings as defined in the *Arm® Glossary*. For example, **IMPLEMENTATION DEFINED**, **IMPLEMENTATION SPECIFIC**, **UNKNOWN**, and **UNPREDICTABLE**.

We recommend the following. If you do not follow these recommendations your system might not work.

Your system requires the following. If you do not follow these requirements your system will not work.

You are at risk of causing permanent damage to your system or your equipment, or of harming yourself.

This information is important and needs your attention.

This information might help you perform a task in an easier, better, or faster way.

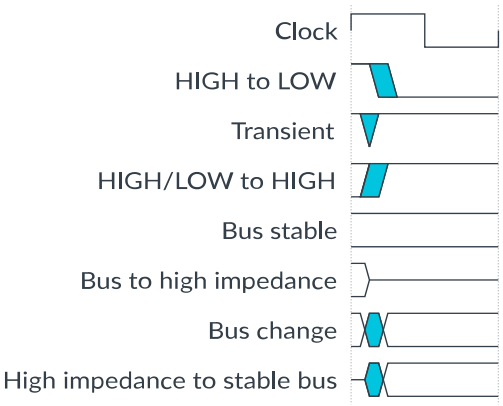
This information reminds you of something important relating to the current content.

Timing diagrams

The following figure explains the components used in timing diagrams. Variations, when they occur, have clear labels. You must not assume any timing information that is not explicit in the diagrams.

Shaded bus and signal areas are undefined, so the bus or signal can assume any value within the shaded area at that time. The actual level is unimportant and does not affect normal operation.

Figure 1: Key to timing diagram conventions



Signals

The signal conventions are:

Signal level

The level of an asserted signal depends on whether the signal is active-HIGH or active-LOW. Asserted means:

- HIGH for active-HIGH signals.
- LOW for active-LOW signals.

Lowercase n

At the start or end of a signal name, n denotes an active-LOW signal.

Useful resources

This document contains information that is specific to this product. See the following resources for other useful information.

Arm documents are available on developer.arm.com/documentation.

Confidential documents are only available to licensees, when logged in. Each document link in the tables below provides direct access to the online version of the document.

Arm product resources	Document ID	Confidentiality
Arm® CoreSight™ Base System Architecture	DEN 0068	Non-Confidential
Arm® CoreSight™ SoC-600 Release Note	109276	Confidential
Arm® CoreSight™ System-on-Chip SoC-600 Configuration and Integration Manual	100807	Confidential

Arm architecture and specifications	Document ID	Confidentiality
AMBA® AHB Protocol Specification	IHI 0033C	Non-Confidential
AMBA® APB Protocol Specification	IHI 0024E	Non-Confidential
AMBA® ATB Protocol Specification	IHI 0032C	Non-Confidential
AMBA® AXI Protocol Specification	IHI 0022K	Non-Confidential
AMBA® AXI-Stream Protocol Specification	IHI 0051B	Non-Confidential
AMBA® Low Power Interface Specification	IHI 0068D	Non-Confidential
Arm® CoreSight™ Architecture Specification v3.0	IHI 0029	Non-Confidential
Arm® Debug Interface Architecture Specification ADIv6.0	IHI 0074D	Non-Confidential
Arm® Embedded Trace Macrocell Architecture Specification ETMv1.0 to ETMv3.5	IHI 0014Q	Non-Confidential
Arm® Embedded Trace Macrocell Architecture Specification ETMv4.0 to ETMv4.3	IHI 0064E	Non-Confidential
CoreSight™ Program Flow Trace Architecture Specification PFTv1.0 and PFTv1.1	IHI 0035	Non-Confidential
Learn the architecture - Realm Management Extension	den0126	Non-Confidential

Non-Arm resources	Document ID	Organization
Accellera IP-XACT version 1685-2009	version 1685-2009	www.accellera.org
IEEE Standard Test Access Port and Boundary Scan Architecture (JTAG)	IEEE 1149.1-2001	https://standards.ieee.org/
Verilog-2001 Standard	IEEE 1364-2001	https://standards.ieee.org/