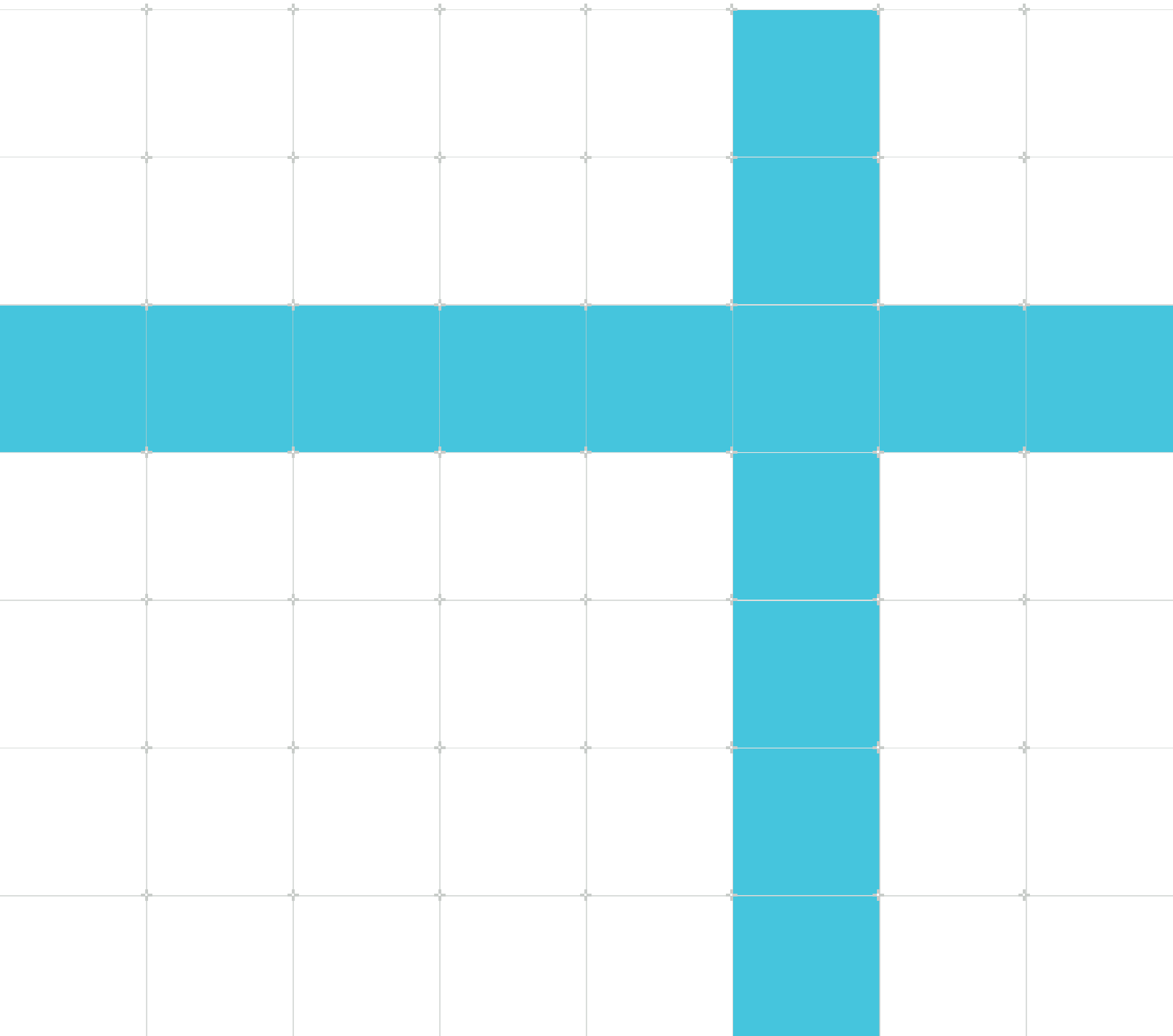
Arm® CoreSight™ ETM-M33

Revision r0p2

Technical Reference Manual

Issue 06
100232_0002_06_en



Arm® CoreSight™ ETM-M33 Technical Reference Manual

This document (100232_0002_06_en) was issued on 2024-08-01. There might be a later issue at http://developer.arm.com/documentation/100232

The product revision is r0p2.

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 book is written for designers of development tools providing support for ETM functionality and hardware and software engineers integrating the macrocell into an ASIC that includes a Cortex®-M33 processor. Implementation-specific behavior is described in this document. You can find complementary information in the *ARM® Embedded Trace Macrocell Architecture Specification ETMv4* and *ARM® Cortex®-M33 Integration and Implementation Manual*.

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. ETM-M33 Functional Description

This part describes ETM-M33 functionality.

1.1 Introduction

This chapter describes ETM-M33.

1.1.1 About the CoreSight ETM-M33

Depending on your implementation, the *Embedded Trace Macrocell* (ETM)-M33 provides non-intrusive program-flow trace for the Cortex®-M33 processor. ETM-M33 generates information that trace software tools use to reconstruct the execution of all or part of a program.

ETM-M33 implements instruction trace only. ETM-M33 is able to trace:

- All instructions, including condition code pass/fail.
- Target addresses of taken direct and indirect branch operations.
- Exceptions.
- Entry to and return from debug state when Halting debug-mode is enabled.
- Cycle counts relating to instruction execution.

ETM-M33 contains resource logic that enables you to control instruction trace. Resource logic includes a single reduced function counter. You specify the exact set of trigger and filter conditions that are required for a particular application.

ETM-M33 is a CoreSight component. For more information about CoreSight, see the *Arm® CoreSight™ Architecture Specification v3.0* and *Arm® CoreSight™ Technology System Design Guide*. For more information about the ETM architecture, see the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* .

1.1.1.1 The CoreSight debug environment

The CoreSight debug environment consists of a software debugger that provides a user interface to ETM-M33. ETM-M33 is configured for instruction trace and it has a single 8-bit ATB interface.

ETM-M33 is designed for use with CoreSight, an extensible system-wide debug and trace architecture from Arm. See the *Arm® CoreSight™ SoC-400 User Guide* for more information about how to test the integration of ETM-M33 in a full CoreSight system.

A software debugger provides a user interface to ETM-M33. You can use this interface to:

- Configure ETM-M33 facilities such as filtering.

- Configure optional trace features such as cycle counting.
- Configure the other CoreSight components such as the *Trace Port Interface Unit* (TPIU).
- Access the processor debug and performance monitor units.

ETM-M33 outputs its trace to the AMBA 4 *Advanced Trace Bus* (ATB) interface.

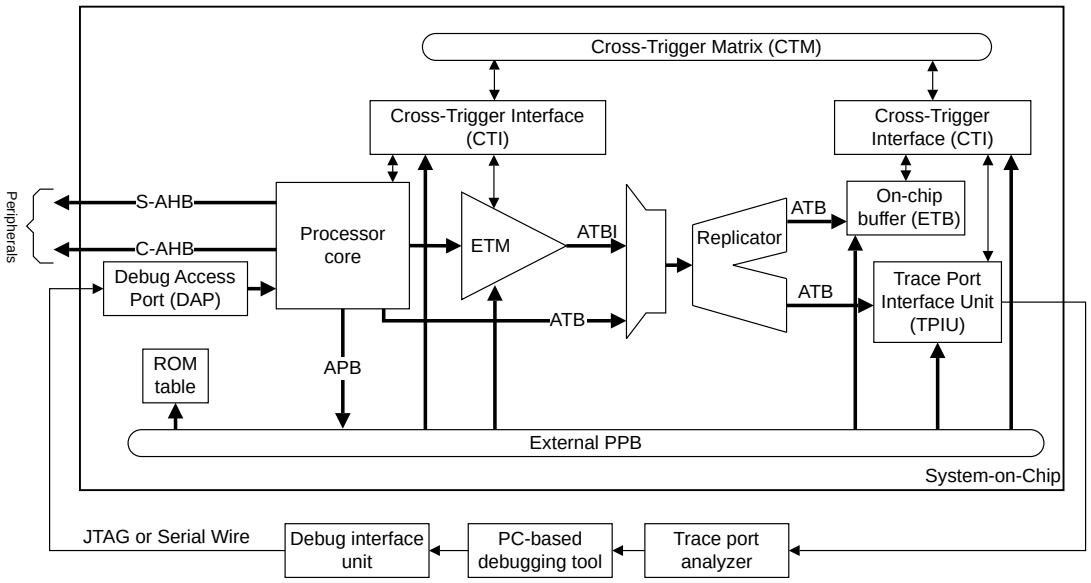
You can use the CoreSight infrastructure to design systems that provide the following options:

- Export the trace information through a trace port. An external *Trace Port Analyzer* (TPA) captures the trace information as the figure in this section shows.
- Write the trace information to a trace-capable device that can access local or system memory. You can read out the trace at low speed using a JTAG or Serial Wire interface.

The debugger has a copy of the executed image from memory and the captured trace information from the TPA or on-chip trace buffer. It decompresses the image to provide full disassembly, with symbols, of the code that was executed. ETM-M33 generates trace information that gives the debugger the capability to link this data back to the original high-level source code. This provides a visualization of how the code was executed on the Cortex®-M33 processor.

The following figure shows an example of how ETM-M33 fits into a CoreSight debug environment to provide instruction trace capabilities in a single processor system. In this example, the external debug software configures the trace and debug components through the *Debug Access Port* (DAP). The top-level ROM table contains a unique identification code for the SoC and the base addresses of the components that are connected to the External PPB on the Cortex®-M33 processor. The ETM-M33 trace interfaces are replicated to provide on-chip storage using the CoreSight ETB or output off-chip using the TPIU. Cross-triggering operates through the *Cross Trigger Interface* (CTI) and *Cross Trigger Matrix* (CTM) components.

Figure 1-1: Example CoreSight system with ETM-M33




Note

The arrows on the thick lines show the transaction direction on buses, from master to slave port. Each bus contains individual signals that go from master to slave and other signals that go from slave to master.

As an alternative to using an external computer to run a software debugger, the Cortex®-M33 processor (or another processor on the SoC) can access ETM-M33 and an on-chip trace buffer to provide self-hosted debug and trace functionality.

1.1.2 Compliance

ETM-M33 is compatible with the CoreSight architecture and implements the ETM v4.2 architecture and complies with AMBA 4 ATB and APB protocols.

This manual complements architecture reference manuals, architecture specifications, protocol specifications, and relevant external standards. It does not duplicate information from these sources.

1.1.2.1 Trace macrocell

ETM-M33 implements the ETM architecture version 4.2.

See the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* .

Related information

Trace Configuration Register on page 28

1.1.2.2 Interconnect architecture

ETM-M33 complies with the *Advanced Microcontroller Bus Architecture* (AMBA) 4 *Advanced Peripheral Bus* (APB) and *Advanced Trace Bus* (ATB) protocols.

See the *AMBA® APB Protocol Version 2.0 Specification* and *Arm® AMBA® 4 ATB Protocol Specification ATBv1.0 and ATBv1.1* .

1.1.3 Features

This section describes the ETM-M33 features that are implementation defined based on either the number of times the feature is implemented, or the size of the feature.

See the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* for information about:

- The trace protocol.
- The features of ETMv4.2.

- Controlling tracing using triggering and filtering resources.

The following table shows the ETM-M33 features that are **IMPLEMENTATION DEFINED**, in terms of either:

- The number of times the feature is implemented.
- The size of the feature.

Table 1-1: ETM-M33 features with implementation-defined number of instances or size

Feature	ETM-M33 configuration	Notes
Address comparators	0 pairs	See bits[3:0] of the 2.1.21 ID Register 4 on page 46.
Data value comparators	0	See bits[7:4] of the 2.1.21 ID Register 4 on page 46.
Context ID comparators	0	See bits[27:24] 2.1.21 ID Register 4 on page 46.
Single-Shot comparator resource	1	The single-shot comparators are only sensitive to the processor comparator inputs.
Counters	1[1]	See bits[30:28] of the 2.1.20 ID Register 3 on page 44.
Cycle count size	12 bits	See bits[28:25] of the 2.1.19 ID Register 2 on page 43.
Sequencer	0	See bits[27:25] of the 2.1.22 ID Register 5 on page 47.
Processor comparator inputs	4	See bits[15:12] of the 2.1.22 ID Register 5 on page 47.
External inputs	4	See bits[8:0] of the 2.1.22 ID Register 5 on page 47.
External outputs	2	-
External input selectors	0	See bits[11:9] of the 2.1.22 ID Register 5 on page 47.
Resource selector pairs	2	See bits[19:16] of the 2.1.21 ID Register 4 on page 46.
Instruction trace port size	8-bit	-
Instruction FIFO	64 byte with 8-bit output	Uses ATB
Claim tag bits	4	-

The following table shows the optional features of the ETM architecture that ETM-M33 implements.

Table 1-2: ETM-M33 implementation of optional features

Feature	Implemented	Notes
Configurable FIFO	No	-
Trace Start/Stop block	Yes	-

[1] Reduced function counter implementation.

Feature	Implemented	Notes
Trace all branches option	Yes	See bit[5] of the 2.1.17 ID Register 0 on page 40.
Trace of conditional instructions	Yes	See bits[13:12] and bit[6] of the 2.1.17 ID Register 0 on page 40.
Cycle counting in instruction trace	Yes	See bit[7] of the 2.1.17 ID Register 0 on page 40.
Data address comparison	No	The Cortex®-M33 processor does not implement data address comparison.
OS Lock mechanism	No	The Cortex®-M33 processor does not implement OS Lock.
Secure non-invasive debug	Yes	The Cortex®-M33 processor implements optional Security Extensions.
Context ID tracing	No	See bits[9:5] of the 2.1.19 ID Register 2 on page 43.
Trace output	Yes	ATB
Timestamp size	64-bit	See bits[28:24] of the 2.1.17 ID Register 0 on page 40.
Memory mapped access to ETM-M33 registers	Yes	See the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* for more information about the Access permissions behaviors on register accesses for different trace unit states.
External debugger access to ETM-M33 registers	Yes	
System instruction access to ETM-M33 registers	No	
VMID comparator support	No	See bits[31:28] of the 2.1.21 ID Register 4 on page 46.
ATB trigger support	Yes	See bit[22] of the 2.1.22 ID Register 5 on page 47.

1.1.4 Interfaces

ETM-M33 has the following main interfaces:

Processor interface

This block connects to the ETM-M33 interface. It tracks the execution information from the processor, decodes the control signals, and passes on the information to the internal interfaces.

ATB

There is one ATB interface which is the instruction ATB interface. This reads single bytes of packet information from the instruction FIFO and sends them over the instruction ATB interface.

APB interface

This block implements the interface to the APB that provides access to the programmable registers.

Production test

This interface contains the scan enable signal used in production testing of ETM-M33.

Related information

Processor interface on page 13
ATB interface on page 13
APB interface on page 14

1.1.5 Configurable options

ETM-M33 has no configurable options.

1.1.6 Test features

ETM-M33 provides the DFTSE input for testing the implemented device.

1.1.7 Design process

ETM-M33 is delivered as synthesizable RTL. Before it can be used in a product, it must go through the following process:

Implementation

For MCUs, often a single design team integrates the processor before synthesizing the complete design. Alternatively, the team can synthesize the processor on its own or partially integrated, to produce a hard macrocell. The hard macrocell is then integrated, possibly by a separate team, which might include integrating RAMs into the design.

Integration

The integrator connects the implemented design into a SoC. This includes connecting it to a memory system and peripherals.

Programming

This is the last process. The system programmer develops the software required to configure and initialize ETM-M33, and tests the required application software.

Each stage of the process:

- Can be performed by a different party.
- Can include implementation and integration choices that affect the ETM-M33 behavior and features.

The operation of the final device depends on:

Build 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 design.

Configuration inputs

The integrator configures some of the ETM-M33 features by tying inputs to specific values. These configurations affect the start-up behavior before any software configuration is made. They can also limit the options available to the software.

Software configuration

The programmer configures ETM-M33 by programming particular values into registers. This affects the ETM-M33 behavior.

This manual refers to **IMPLEMENTATION DEFINED** features that are applicable to build configuration options. Reference to a feature that is included means that the appropriate build and pin configuration options are selected. Reference to an enabled feature means one that has also been configured by software.

1.1.8 Documentation

ETM-M33 documentation is as follows:

Technical Reference Manual

The *Technical Reference Manual* (TRM) describes the functionality and the effects of functional options on the ETM-M33 behavior. It is required at all stages of the design flow. Some behavior described in the TRM might not be relevant because of the way that ETM-M33 is implemented and integrated.

Integration and Implementation Manual

For both the processor and ETM-M33, the *Arm® Cortex®-M33 Integration and Implementation Manual* (IIM) describes:

- The available build configuration options and related issues in selecting them.
- How to configure the *Register Transfer Level* (RTL) with the build configuration options.
- How to integrate the processor into a SoC. This includes a description of the pins that the integrator must tie off to configure the macrocell for the required integration.
- The processes to sign off the integration and implementation of the design.

The Arm product deliverables include reference scripts and information about using them to implement your design. Reference methodology flows supplied by Arm are example reference implementations. Contact your EDA vendor for EDA tool support.

Reference methodology documentation from your EDA tools vendor complements the IIM.

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

1.1.9 Product revisions

This section describes the difference in functionality between product revisions in ETM-M33.

r0p0 First release.

r0p1

The following changes that are made in this release are:

- TRCAUTHSTATUS.SNID bitfield is updated to:
  - Include `0b00` value, indicating that Non-Invasive Debug is not implemented.
  - RAZ when the configuration of the processor does not include ARMv8-M Security Extension.
- Various engineering errata fixes.

r0p2

Various engineering errata fixes.

1.2 Functional Description

This chapter describes the ETM-M33 functional features and operation.

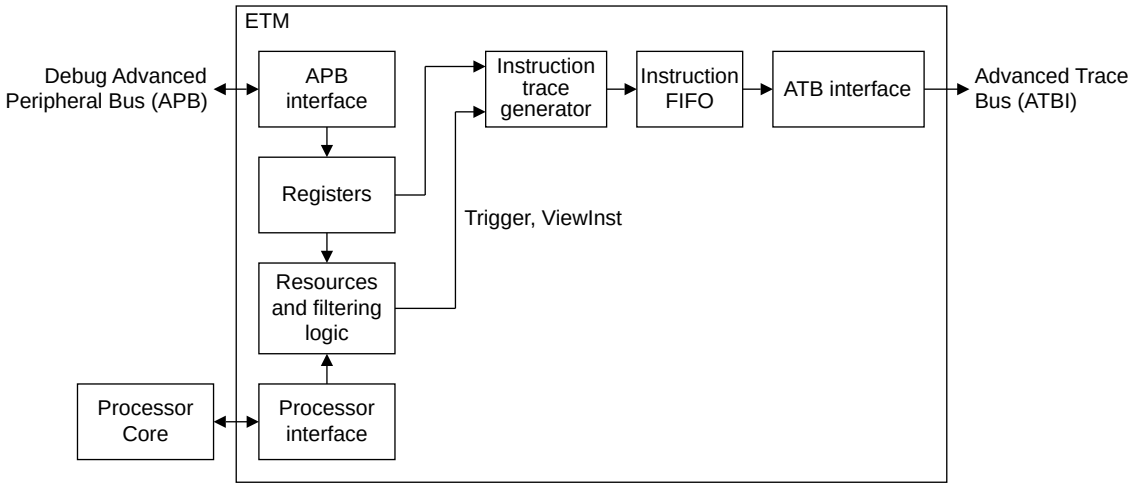
1.2.1 About the functions

ETM-M33 performs real-time instruction tracing based on the ETM architecture ETMv4.2.

ETM-M33 is a CoreSight component and is an integral part of the Arm Real-time Debug solution, DS-5 Development Studio.

The following figure shows the main functional blocks of ETM-M33.

Figure 1-2: ETM-M33 block diagram



1.2.1.1 Processor interface

This block connects to the Cortex®-M33 core. It tracks the execution information from the core, decodes the control signals, and passes on the information to the internal interfaces.

1.2.1.2 Instruction trace generator

This block generates the trace packets that are a compressed form of the instruction execution information provided by the Cortex®-M33 processor. The trace packets are then passed to the instruction FIFO.

1.2.1.3 FIFO

This block buffers bursts of trace packets. There is one FIFO provided for the instruction trace stream.

1.2.1.4 Resources and filtering logic

These blocks contain resources which are programmed by trace software to trigger and filter the trace information. They start and stop trace generation, depending on the conditions that have been set.

1.2.1.5 ATB interface

There is one ATB interface in Cortex®-M33.

Instruction ATB interface

This reads single bytes of packet information from the instruction FIFO and sends them over the instruction ATB interface.

1.2.1.6 APB interface

This block implements the interface to the APB that provides access to the programmable registers.

1.2.1.7 Global timestamping

ETM-M33 supports connection to a global timestamp source.

This provides a 64-bit timestamp that a debugger can use for coarse-grained profiling and correlation of the trace source. Arm recommends that the timestamp counter is no slower than 10% of the processor clock frequency.

1.2.2 ETMEVENT connectivity

This section describes how the ETM-M33 ETMEVENT inputs and outputs are connected to the CTI.

The following table shows the connection of the ETMEVENTS inputs that come from the CTI.

Table 1-3: ETMEVENTS connections

Bits	Description
[3:0]	CTI Trigger Output

The following table shows the ETM-M33 output resources, ETMEVENTM, connected to the CTI.

Table 1-4: ETMEVENTM connections to CTI

ETM-M33 output	CTI input
ETM External Output 0	CTI Trigger Input 4
ETM External Output 1	CTI Trigger Input 5

Related information

Configurable options on page 10

1.2.3 Operation

This section describes the ETM-M33 **IMPLEMENTATION DEFINED** features. These features are **IMPLEMENTATION DEFINED**.

For information on the operation, see the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4*.

Related information

Programmers Model on page 18

1.2.3.1 Implementation defined registers

There are two groups of ETM-M33 registers:

- Registers that are completely defined by the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4*.
- Registers that are partly **IMPLEMENTATION DEFINED**.

1.2.3.2 Precise ViewInst events

The only condition that ensures ViewInst is precise is that the enabling event condition is TRUE.

For more information, see the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* .

1.2.3.3 Parallel instruction execution

The Cortex®-M33 processor supports parallel instruction execution. This means the macrocell is capable of tracing two instructions per cycle.

If ViewInst is active for a cycle, the ETM-M33 always traces the first of any paired instructions.

1.2.3.4 Trace features

The ETM-M33 implements the following ETMv4.2 trace features:

- Cycle-accurate tracing.
- Timestamping.

See the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* for descriptions of these features.

1.2.3.5 Packet formats

This section describes the packet formats that the ETM-M33 instruction trace interface supports.

The ETM-M33 instruction trace interface does not support the following trace packet types:

- Speculation resolution packets are not supported.
- Conditional tracing packets.
- Q instruction trace packets are not supported.

See the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* for the trace packet format descriptions.

1.2.3.6 Resource selection

The ETM-M33 uses event selectors to control resources.

The ETM-M33 controls the following resources:

- Trace events, triggers, and markers in the trace stream.
- Timestamp event.
- ViewInst event.
- Reduced function counter.

An event selector is configured to be sensitive to a resource selector pair, and one resource selector pair can control more than one event selector. The event selectors for Cortex®-M33 are located in registers TRCEVENTCTL0R, TRCCNTCTLRn, TRCTSCTLR, and TRCVICTLR.

The ETM-M33 provides one fixed resource selector pair, registers TRCRSCTLR0 and TRCRSCTLR1 with respective static values of 0 = FALSE and 1 = TRUE, and one configurable resource selector pair, registers TRCRSCTLR2 and TRCRSCTLR3. A resource selector pair enables up to two resource groups to be selected, and enables one or more resources to be selected in each group. If more than one resource is selected, the outputs of the selected resources are OR-gated.

See the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* for more information.

The following table shows the resources that can be selected for the Instruction trace.

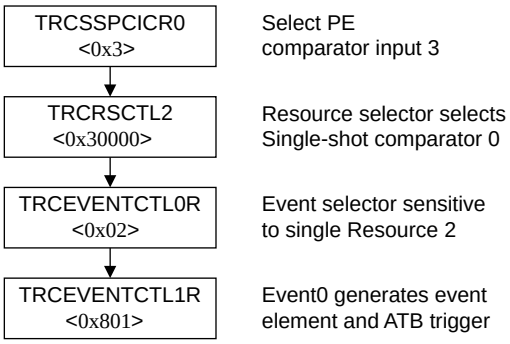
Table 1-5: Instruction trace resource selection

Group	Select	Resource
0b0000	0-3	External input selectors 0-3
0b0001	0-3	Inputs from Processor DWT comparators Element 0-3
0b0010	0	Counter at zero 0
0b0011	0	Single-Shot Comparator control 0
0b0100-0b1111	0-15	Reserved

For example, the following figure shows the steps necessary to use a Single-Shot comparator to generate a trigger event and an ATB trigger. This example uses the user-configurable resource selector 2.

The *Data Watchpoint and Trace* (DWT) unit in the processor control the processor comparator inputs.

Figure 1-3: Trigger event resource selection



The ETM-M33 single shot and start-stop logic, when present, might not reliably trigger where DWT comparators are programmed for comparisons other than instruction address comparison.

1.2.3.7 Trace flush behavior

The ETM-M33 observes events that can be confirmed to have reached the trace bus output with the use of the ATB flush protocol. The ATB port must be flushed when an instruction trace is required.

The ETM-M33 internally flushes instruction trace whenever the flush request is seen. When the processor enters a low power state, this also causes instruction trace to be output from the ETM-M33.

1.2.3.8 Low-power state behavior

When the processor enters a low-power state, there is a delay before the resources to the ETM-M33 become inactive.

This permits the last instruction executed, to trigger a comparator, or update the counter, and the resultant event packet to be inserted in the specified trace stream. This event packet is presented on the trace bus before the ETM-M33 itself enters a low-power state.

If an event packet is generated for a different reason, it is not guaranteed to be output before the ETM-M33 enters a low-power state, but is traced when the processor leaves the low-power state, if the ETM-M33 logic is not reset before this can occur.

This low-power behavior can be disabled using TRCEVENTCTL1R.LPOVERRIDE bit, see 2.1.6 Event Control 0 Register on page 29. In this case, the ETM-M33 resources remain active.

1.2.3.9 Cycle counter

The ETM-M33 uses a 12-bit cycle counter.

It does not count when non-invasive debug is disabled or when the processor is in a low-power state.

1.2.3.10 Event tracing and triggers

Instruction event packets can be inserted in the instruction trace stream on every cycle, but if events are traced continuously on every cycle the instruction FIFO is unable to drain and overflows.

When used with the optimized Cortex®-M33 TPIU, ATB triggers must not be enabled, TRCEVENTCTL1R.ATB must be set to 0.


Note

ATB triggers must not be enabled for any *Embedded Trace Macrocell* (ETM) with the Cortex®-M33 TPIU.

1.3 Programmers Model

This chapter describes the programmers model.

1.3.1 About the programmers model

This chapter describes the mechanisms for programming the registers used to set up the trace and triggering facilities of the macrocell. The programmers model enables you to use the ETM-M33 registers to control the macrocell.

1.3.2 Modes of operation and execution

This section describes how to control the ETM programming and read and program the ETM registers.

1.3.2.1 Controlling ETM-M33 programming

When programming the ETM-M33 registers, you must enable all the changes at the same time.

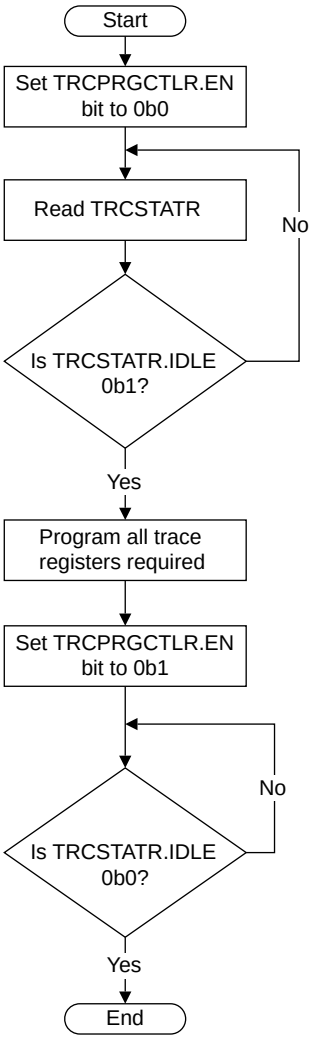
For example, if the counter is reprogrammed, it might start to count based on incorrect events, before the trigger condition has been correctly set up.

To disable instruction trace operations during programming, use:

- The trace program enable bit in the TRCPRGCTLR.
- The TRCSTATR to indicate the ETM-M33 status.

The following figure shows the procedure to use.

Figure 1-4: Programming ETM registers



The Cortex®-M33 processor does not have to be in debug state while you program the ETM-M33 registers.

Related information

Programming Control Register on page 26
Status Register on page 27

1.3.2.2 Programming and reading ETM registers

To access the ETM registers, use the external APB interface. This provides a direct method of programming the ETM.

Related information

Programming Control Register on page 26

2. ETM-M33 Register Descriptions

This part describes the ETM-M33 system registers.

2.1 ETM-M33 registers

This section summarizes the ETM-M33 registers.

2.1.1 Register summary

This section summarizes the ETM-M33 registers.

For full descriptions of the ETM-M33 registers, see:

- The *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* , for registers not described in this document.

Table 2-1: ETM-M33 register summary on page 22 lists the ETM-M33 registers in numerical order and describes each register.

The register table includes additional information about each register:

- The register access type. This is read-only, write-only, or read and write.
- The base offset address of the register. The base offset of a register is always four times its register number. For information on the base address of the registers, see *Arm®v8-M Architecture Reference Manual*.
- Additional information about the implementation of the register, where appropriate.

- Registers not listed here are not implemented. Reading a non-implemented register address returns 0. Writing to a non-implemented register address has no effect.
- In the following table:
  - The Reset value column shows the value of the register immediately after an ETM-M33 reset. For read-only registers, every read of the register returns this value.
  - Access type is described as follows:

RW	Read and write.
RO	Read only.
WO	Write only.

All the ETM-M33 registers are 32 bits wide.

Table 2-1: ETM-M33 register summary

Register number	Base offset	Name	Type	Reset value	Description
1	0x004	TRCPRGCTLR	RW	0x00000000	2.1.2 Programming Control Register on page 26
3	0x00C	TRCSTATR	RO	-	2.1.3 Status Register on page 27
4	0x010	TRCCONFIGR	RW	-	2.1.4 Trace Configuration Register on page 28
6	0x018	TRCAUXCTLR[2]	RW	-	UNK/SBZP.
8	0x020	TRCEVENTCTL0R	RW	-	2.1.6 Event Control 0 Register on page 29
9	0x024	TRCEVENTCTL1R	RW	-	2.1.7 Event Control 1 Register on page 30
11	0x02C	TRCSTALLCTLR	RW	-	2.1.8 Stall Control Register on page 31
12	0x030	TRCTSCTLR	RW	-	2.1.9 Global Timestamp Control Register on page 32
13	0x034	TRCSYNCPR	RO	0xA	2.1.10 Synchronization Period Register on page 33
14	0x038	TRCCCCTLR	RW	-	2.1.11 Cycle Count Control Register on page 34
16	0x040	TRCTRACEIDR	RW	-	2.1.12 Trace ID Register on page 34
32	0x080	TRCVICTLR	RW	-	2.1.13 ViewInst Main Control Register on page 35
80-81	0x140	TRCCNTRLDVR0	RW	-	2.1.14 Counter Reload Value Registers 0 on page 37
96	0x180	TRCIDR8	RO	0x00000000	2.1.15 ID Register 8-13 on page 37
97	0x184	TRCIDR9	RO	0x00000000	
98	0x188	TRCIDR10	RO	0x00000000	
99	0x18C	TRCIDR11	RO	0x00000000	
100	0x190	TRCIDR12	RO	0x00000001	
101	0x194	TRCIDR13	RO	0x00000000	
112	0x1C0	TRCIMSPEC0	RW	0x00000000	2.1.16 Implementation Specific Register 0 on page 40
120	0x1E0	TRCIDR0	RO	0x280006E1	2.1.17 ID Register 0 on page 40
121	0x1E4	TRCIDR1	RO	0x4100F421	2.1.18 ID Register 1 on page 42
122	0x1E8	TRCIDR2	RO	0x00000004	2.1.19 ID Register 2 on page 43
123	0x1EC	TRCIDR3	RO	0x0F090004	2.1.20 ID Register 3 on page 44
124	0x1F0	TRCIDR4	RO	-	2.1.21 ID Register 4 on page 46
125	0x1F4	TRCIDR5	RO	0x90C70004	2.1.22 ID Register 5 on page 47
130-143	0x208-0x20C	TRCRSCTLR2-3	RW	-	2.1.23 Resource Selection Registers 2-3 on page 48
160	0x280	TRCSSCCR0	RW	-	2.1.24 Single-shot Comparator Control Register 0 on page 49
168	0x2A0	TRCSSCSR0	RW	-	2.1.25 Single-shot Comparator Status Register 0 on page 50
176	0x2C0	TRCSSPCICR0	RW	-	2.1.26 Single-shot Processor Comparator Input Control Register on page 51
196	0x310	TRCPDCR	RW	0x00000000	2.1.27 Power Down Control Register on page 52
197	0x314	TRCPDSR	RO	0x00000003	2.1.28 Power Down Status Register on page 53

[2] The processor does not implement this register.
[3] **0x00114000** For 4 comparator configuration.
0x00112000 For 2 comparator configuration.

Register number	Base offset	Name	Type	Reset value	Description
953	0xEE4	TRCITATBIDR	RW	-	2.1.29.1 Integration ATB Identification Register on page 54
957	0xEF4	TRCITIATBINR	RO	-	2.1.29.2 Integration Instruction ATB In Register on page 55
959	0xEFC	TRCITIATBOUTR	RW	-	2.1.29.3 Integration Instruction ATB Out Register on page 56
960	0xF00	TRCITCTRL	RW	0x00000000	2.1.29.4 Integration Mode Control Register on page 57
1000	0xFA0	TRCCLAIMSET	RW	0x0000000F	2.1.30 Claim Tag Set Register on page 58
1001	0xFA4	TRCCLAIMCLR	RW	0x00000000	2.1.31 Claim Tag Clear Register on page 58
1006	0xFB8	TRCAUTHSTATUS	RO	-	2.1.32 Authentication Status Register on page 59
1007	0xFBC	TRCDEVARCH	RO	0x47724A13	2.1.33 Device Architecture Register on page 60
1010	0xFC8	TRCDEVID	RO	0x00000000	2.1.34 Device ID Register on page 61
1011	0xFCC	TRCDEVTYPE	RO	0x00000013	2.1.35 Device Type Register on page 62
1012-1019	0xFD0-0xFEC	TRCPIDR4-7, TRCPIDR0-3	RO	-	2.1.36 Peripheral Identification Registers on page 63
1020-1023	0xFF0-0xFFC	TRCCIDR0-3	RO	-	2.1.37 Component Identification Registers on page 64

2.1.1.1 General control and ID registers

The following table shows the general control and ID registers in numerical order.

Table 2-2: General control and ID registers

Register number	Name	Base offset	Description
1	TRCPRGCTLR	0x004	2.1.2 Programming Control Register on page 26
3	TRCSTATR	0x00C	2.1.3 Status Register on page 27
4	TRCCONFIGR	0x010	2.1.4 Trace Configuration Register on page 28
4	TRCAUXCTLR	0x018	2.1.5 Auxiliary Control Register on page 29
8	TRCEVENTCTL0R	0x020	2.1.6 Event Control 0 Register on page 29
9	TRCEVENTCTL1R	0x024	2.1.7 Event Control 1 Register on page 30
11	TRCSTALLCTLR	0x02C	2.1.8 Stall Control Register on page 31
12	TRCTSCTLR	0x030	2.1.9 Global Timestamp Control Register on page 32
13	TRCSYNCPR	0x034	2.1.10 Synchronization Period Register on page 33
14	TRCCCCTLR	0x038	2.1.11 Cycle Count Control Register on page 34
16	TRCTRACEIDR	0x040	2.1.12 Trace ID Register on page 34

2.1.1.2 Trace filtering control register

The following table shows the trace filtering control register.

Table 2-3: Trace filtering control register

Register number	Name	Base offset	Description
32	TRCVICTLR	0x080	2.1.13 ViewInst Main Control Register on page 35

2.1.1.3 Derived resource register

The following table shows the derived resource register.

Table 2-4: Derived resource register

Register number	Name	Base offset	Description
80-81	TRCCNTRLDVR0	0x140	2.1.14 Counter Reload Value Registers 0 on page 37

2.1.1.4 Implementation specific and identification registers

The following table shows the **IMPLEMENTATION SPECIFIC** and identification registers in numerical order.

Table 2-5: Implementation specific and identification registers

Register number	Name	Base offset	Description
96	TRCIDR8	0x180	2.1.15 ID Register 8-13 on page 37
97	TRCIDR9	0x184	
98	TRCIDR10	0x188	
99	TRCIDR11	0x18C	
100	TRCIDR12	0x190	
101	TRCIDR13	0x194	
112	TRCIMSPEC0	0x1C0	2.1.16 Implementation Specific Register 0 on page 40
120	TRCIDR0	0x1E0	2.1.17 ID Register 0 on page 40
121	TRCIDR1	0x1E4	2.1.18 ID Register 1 on page 42
122	TRCIDR2	0x1E8	2.1.19 ID Register 2 on page 43
123	TRCIDR3	0x1EC	2.1.20 ID Register 3 on page 44
124	TRCIDR4	0x1F0	2.1.21 ID Register 4 on page 46
125	TRCIDR5	0x1F4	2.1.22 ID Register 5 on page 47

2.1.1.5 Resource selection register

The following table shows the resource selection register.

Table 2-6: Resource selection register

Register number	Name	Base offset	Description
130-143	TRCRSCTLR2-3	0x208-0x20C	2.1.23 Resource Selection Registers 2-3 on page 48

2.1.1.6 Single-shot comparator registers

The following table shows the single-shot comparator registers in numerical order.

Table 2-7: Single-shot comparator registers

Register number	Name	Base offset	Description
160	TRCSSCCR0	0x280	2.1.24 Single-shot Comparator Control Register 0 on page 49
168	TRCSSCSR0	0x2A0	2.1.25 Single-shot Comparator Status Register 0 on page 50
176	TRCSSPCICR0	0x2C0	2.1.26 Single-shot Processor Comparator Input Control Register on page 51

2.1.1.7 Power control registers

The following table shows the power control registers in numerical order.

Table 2-8: Power control registers

Register number	Name	Base offset	Description
196	TRCPDCR	0x310	2.1.27 Power Down Control Register on page 52
197	TRCPDSR	0x314	2.1.28 Power Down Status Register on page 53

2.1.1.8 Integration test registers

The following table shows the integration test registers in numerical order.

Table 2-9: Integration test registers

Register number	Name	Base offset	Description
953	Integration ATB Identification Register	0xEE4	2.1.29.1 Integration ATB Identification Register on page 54
957	Integration Instruction ATB In Register	0xEF4	2.1.29.2 Integration Instruction ATB In Register on page 55
959	Integration Instruction ATB Out Register	0xEFC	2.1.29.3 Integration Instruction ATB Out Register on page 56

2.1.1.9 CoreSight management registers

The following table shows the CoreSight management registers in numerical order.

Table 2-10: CoreSight management registers

Register number	Name	Base offset	Description
960	TRCITCTRL	0xF00	2.1.29.4 Integration Mode Control Register on page 57
1000	TRCCLAIMSET	0xFA0	2.1.30 Claim Tag Set Register on page 58
1001	TRCCLAIMCLR	0xFA4	2.1.31 Claim Tag Clear Register on page 58
1006	TRCAUTHSTATUS	0xFB8	2.1.32 Authentication Status Register on page 59
1007	TRCDEVARCH	0xFBC	2.1.33 Device Architecture Register on page 60
1010	TRCDEVID	0xFC8	2.1.34 Device ID Register on page 61
1011	TRCDEVTYPE	0xFCC	2.1.35 Device Type Register on page 62
1012-1019	TRCPIDR0-7	0xFD0-0xFEC	2.1.36 Peripheral Identification Registers on page 63
1020-1023	TRCCIDR0-3	0xFF0-0xFFC	2.1.37 Component Identification Registers on page 64

2.1.2 Programming Control Register

The TRCPRGCTLR enables ETM-M33.

Usage constraints

See 1.3.2.1 Controlling ETM-M33 programming on page 18.

Configurations

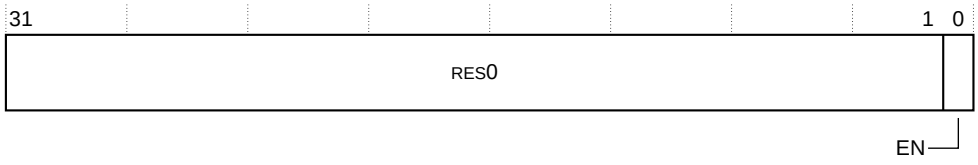
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-2: General control and ID registers on page 23.

The following figure shows the TRCPRGCTLR bit assignments.

Figure 2-1: TRCPRGCTLR bit assignments



The following table shows the TRCPRGCTLR bit assignments.

Table 2-11: TRCPRGCTLR bit assignments

Bits	Name	Function
[31:1]	-	**RES0**.

Bits	Name	Function
[0]	EN	Trace unit enable bit: **0** The trace unit is disabled. **1** The trace unit is enabled.

Related information

Controlling ETM-M33 programming on page 18
Programming and reading ETM registers on page 20

2.1.3 Status Register

The TRCSTATR indicates the ETM-M33 status.

Usage constraints

There are no usage constraints.

Configurations

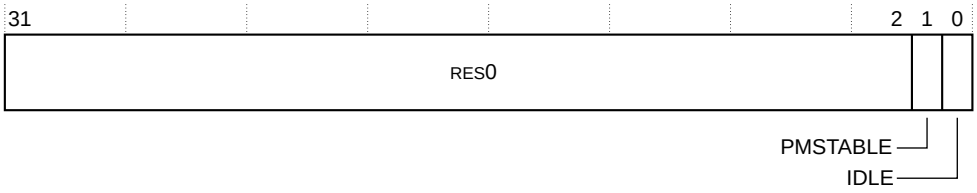
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-2: General control and ID registers on page 23.

The following figure shows the TRCSTATR bit assignments.

Figure 2-2: TRCSTATR bit assignments



The following table shows the TRCSTATR bit assignments.

Table 2-12: TRCSTATR bit assignments

Bits	Name	Function
[31:2]	-	**RES0**.
[1]	PMSTABLE	Indicates whether the ETM-M33 registers are stable and can be read: **0** The programmers model is not stable. **1** The programmers model is stable.
[0]	IDLE	Indicates that the trace unit is inactive: **0** ETM-M33 is not idle. **1** ETM-M33 is idle.

2.1.4 Trace Configuration Register

The TRCCONFIGR sets the basic tracing options for the trace unit.

Usage constraints

This register must always be programmed as part of the trace unit initialization.

Only accepts writes when the trace unit is disabled.

Configurations

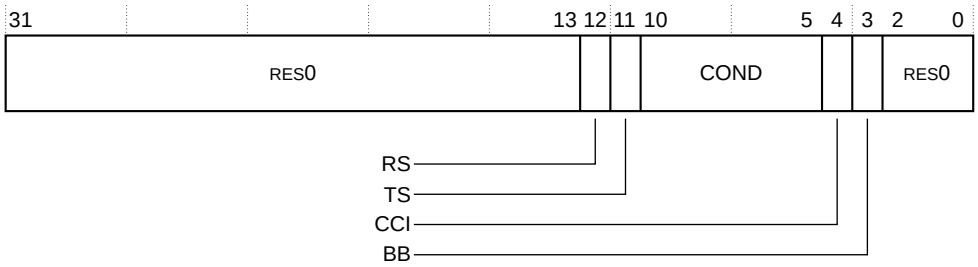
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-2: General control and ID registers on page 23.

The following figure shows the TRCCONFIGR bit assignments.

Figure 2-3: TRCCONFIGR bit assignments



The following table shows the TRCCONFIGR bit assignments.

Table 2-13: TRCCONFIGR bit assignments

Bits	Name	Function
[31:13]	-	**RES0**.
[12]	RS	Return stack enable: **0** Return stack disabled. **1** Return stack enabled.
[11]	TS	Global timestamp tracing: **0** Global timestamp tracing disabled. **1** Global timestamp tracing enabled. For more global timestamping options, see 2.1.9 Global Timestamp Control Register on page 32.

Bits	Name	Function
[10:5]	COND	Conditional instruction tracing. The supported values are: **0b0000** Conditional instruction tracing is disabled. **0b0001** Conditional load instructions are traced. **0b0010** Conditional store instructions are traced. **0b0011** Conditional load and store instructions are traced. **0b0111** All conditional instructions are traced. All other values are Reserved.
[4]	CCI	Cycle counting in instruction trace: **0** Cycle counting in instruction trace disabled. **1** Cycle counting in instruction trace enabled. For more cycle counting options, see 2.1.11 Cycle Count Control Register on page 34.
[3]	BB	Branch broadcast mode: **0** Branch broadcast mode disabled. **1** Branch broadcast mode enabled.
[2:0]	-	**RES0.**

2.1.5 Auxiliary Control Register

The processor does not implement TRCAUXCTLR, so this register is always UNK/SBZP.

2.1.6 Event Control 0 Register

The TRCEVENTCTL0R controls the tracing of events in the trace stream. The events also drive the ETM-M33 external outputs.

Usage constraints

This register must always be programmed as part of the trace unit initialization.

Only accepts writes when the trace unit is disabled.

Configurations

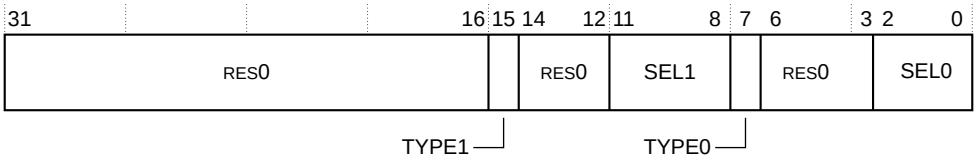
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-2: General control and ID registers on page 23.

The following figure shows the TRCEVENTCTL0R bit assignments.

Figure 2-4: TRCEVENTCTL0R bit assignments



The following table shows the TRCEVENTCTL0R bit assignments.

Table 2-14: TRCEVENTCTL0R bit assignments

Bits	Name	Function
[31:16]	-	**RES0.**
[15]	TYPE1	Selects the resource type for event 1: **0** Single selected resource. **1** Boolean combined resource pair.
[14:12]	-	**RES0.**
[11:8]	SEL1	Selects the resource number, based on the value of TYPE1: When TYPE1 is 0, selects a single selected resource from 0-15 defined by SEL1[2:0]. When TYPE1 is 1, selects a Boolean combined resource pair from 0-7 defined by SEL1[2:0].
[7]	TYPE0	Selects the resource type for event 0: **0** Single selected resource. **1** Boolean combined resource pair.
[6:3]	-	**RES0.**
[2:0]	SEL0	Selects the resource number, based on the value of TYPE0: When TYPE0 is 0, selects a single selected resource from 0-15 defined by SEL0[2:0]. When TYPE0 is 1, selects a Boolean combined resource pair from 0-7 defined by SEL0[2:0].

2.1.7 Event Control 1 Register

The TRCEVENTCTL1R controls how the events selected by TRCEVENTCTL0R behave.

See 2.1.6 Event Control 0 Register on page 29.

Usage constraints

This register must always be programmed as part of the trace unit initialization.

Only accepts writes when the trace unit is disabled.

Configurations

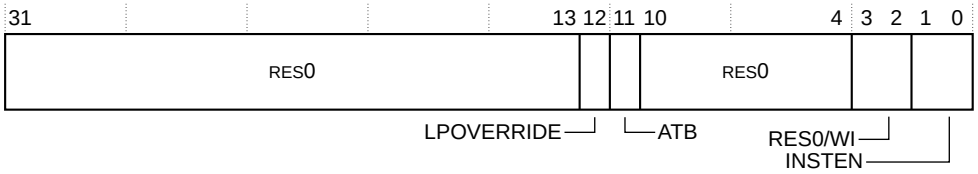
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-2: General control and ID registers on page 23.

The following figure shows the TRCEVENTCTL1R bit assignments.

Figure 2-5: TRCEVENTCTL1R bit assignments



The following table shows the TRCEVENTCTL1R bit assignments.

Table 2-15: TRCEVENTCTL1R bit assignments

Bits	Name	Function
[31:13]	-	**RES0.**
[12]	LPOVERRIDE	Low power state behavior override: **0** Low power state behavior unaffected. **1** Low power state behavior overridden. The resources and Event trace generation are unaffected by entry to a low power state.
[11]	ATB	ATB trigger enable: **0** ATB trigger disabled. **1** ATB trigger enabled.
[10:4]	-	**RES0**
[3:2]	-	**RES0**/WI
[1:0]	INSTEN	One bit per event, to enable generation of an event element in the instruction trace stream when the selected event occurs: **0** Event does not cause an event element. **1** Event causes an event element.

2.1.8 Stall Control Register

The TRCSTALLCTLR enables ETM-M33 to stall the processor if the ETM-M33 FIFO goes over the programmed level to minimize risk of overflow.

Usage constraints

Only accepts writes when the trace unit is disabled.

This register must always be programmed as part of the trace unit initialization.

Configurations

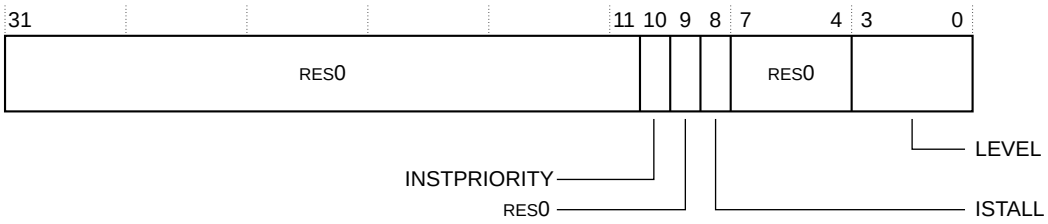
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-2: General control and ID registers on page 23.

The following figure shows the TRCSTALLCTLR bit assignments.

Figure 2-6: TRCSTALLCTLR bit assignments



The following table shows the TRCSTALLCTLR bit assignments.

Table 2-16: TRCSTALLCTLR bit assignments

Bits	Name	Function
[31:11]	-	**RES0**.
[10]	INSTPRIORITY	Prioritize instruction trace if instruction trace buffer space is less than LEVEL: **0** The trace unit must not prioritize instruction trace. **1** The trace unit can prioritize instruction trace.
[9]	-	**RES0**.
[8]	ISTALL	Stall processor based on instruction trace buffer space: **0** The trace unit must not stall the processor. **1** The trace unit can stall the processor.
[7:4]	-	**RES0**.
[3:0]	LEVEL	Threshold at which stalling becomes active. This provides four levels. This level can be varied to optimize the level of invasion caused by stalling, balanced against the risk of a FIFO overflow: **0b0000** Zero invasion. This setting has a greater risk of a FIFO overflow **0b1111** Maximum invasion occurs but there is less risk of a FIFO overflow.

2.1.9 Global Timestamp Control Register

The TRCTSCTLR controls the insertion of global timestamps into the trace stream. A timestamp is always inserted into the instruction trace stream.

Usage constraints

Only accepts writes when the trace unit is disabled.

This register must always be programmed as part of the trace unit initialization.

Configurations

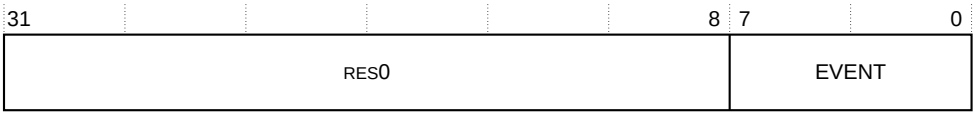
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-2: General control and ID registers on page 23.

The following figure shows the TRCTSCTLR bit assignments.

Figure 2-7: TRCTSCTLR bit assignments



The following table shows the TRCTSCTLR bit assignments.

Table 2-17: TRCTSCTLR bit assignments

Bits	Name	Function
[31:8]	-	**RES0.**
[7:0]	EVENT	An event selector. When the selected event is triggered, the trace unit inserts a global timestamp into the trace streams.

2.1.10 Synchronization Period Register

The TRCSYNCPR specifies the period of trace synchronization of the trace streams. TRCSYNCPR defines a number of bytes of trace between requests for trace synchronization. This value is always a power of two.

Usage constraints

The register is implemented as RO and the synchronization period, given in PERIOD, is `0b01010`.

Configurations

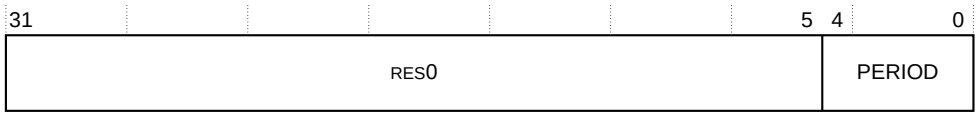
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-2: General control and ID registers on page 23.

The following figure shows the TRCSYNCPR bit assignments.

Figure 2-8: TRCSYNCPR bit assignments



The following table shows the TRCSYNCPR bit assignments.

Table 2-18: TRCSYNCPR bit assignments

Bits	Name	Function
[31:5]	-	**RES0.**
[4:0]	PERIOD	Defines the number of bytes of trace between trace synchronization requests as a total of the number of bytes generated by the instruction stream. The number of bytes is 2^{10} where 10 is the value of this field.

2.1.11 Cycle Count Control Register

The TRCCCCTLR sets the threshold value for instruction trace cycle counting. The threshold represents the minimum interval between cycle count trace packets.

Usage constraints

Only accepts writes when the trace unit is disabled.

This register must always be programmed as part of the trace unit initialization.

Configurations

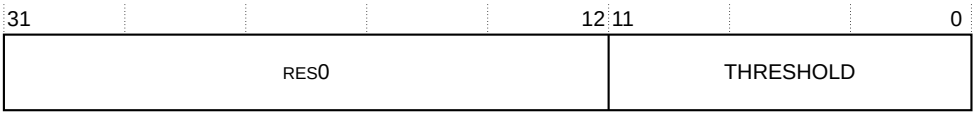
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-2: General control and ID registers on page 23.

The following figure shows the TRCCCCTLR bit assignments.

Figure 2-9: TRCCCCTLR bit assignments



The following table shows the TRCCCCTLR bit assignments.

Table 2-19: TRCCCCTLR bit assignments

Bits	Name	Function
[31:12]	-	**RES0.**
[11:0]	THRESHOLD	Instruction trace cycle count threshold.

2.1.12 Trace ID Register

The TRCTRACEIDR sets the trace ID on the trace bus.

Usage constraints

In a CoreSight system, writing of reserved trace ID values, `0x00` and `0x70-0x7F`, is **UNPREDICTABLE**. This register must always be programmed as part of the trace unit initialization.

Only accepts writes when the trace unit is disabled.

Configurations

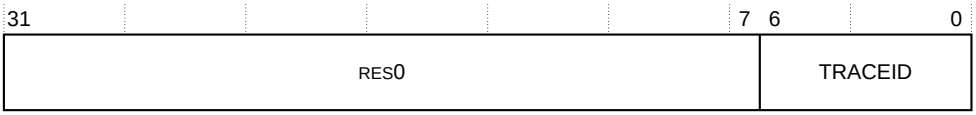
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-2: General control and ID registers on page 23.

The following figure shows the TRCTRACEIDR bit assignments.

Figure 2-10: TRCTRACEIDR bit assignments



The following table shows the TRCTRACEIDR bit assignments.

Table 2-20: TRCTRACEIDR bit assignments

Bits	Name	Function
[31:7]	-	**RES0.**
[6:0]	TRACEID	Trace ID value. This provides the instruction trace ID.

2.1.13 ViewInst Main Control Register

The TRCVICTLR controls instruction trace filtering.

Usage constraints

Only accepts writes when the trace unit is disabled.

Only returns stable data when TRCSTATR.PMSTABLE is 1.

Must be programmed, particularly to set the value of the SSSTATUS bit, that sets the state of the start/stop logic.

Configurations

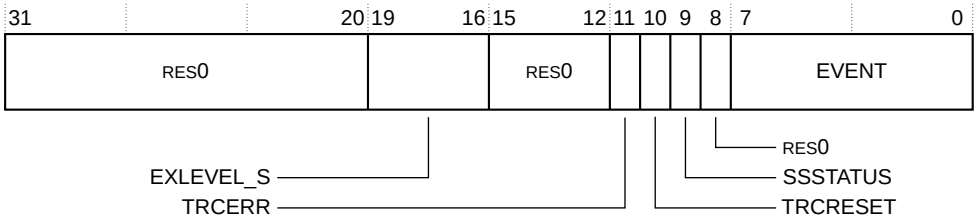
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-3: Trace filtering control register on page 24.

The following figure shows the TRCVICTLR bit assignments.

Figure 2-11: TRCVICTLR bit assignments



The following table shows the TRCVICTLR bit assignments.

Table 2-21: TRCVICTLR bit assignments

Bits	Name	Function
[31:20]	-	**RES0.**
[19:16]	EXLEVEL_S	In Secure state, each bit controls whether instruction tracing is enabled for the corresponding Exception level: **0** The trace unit generates instruction trace, in Secure state, for Exception level n. **1** The trace unit generates instruction trace, in Non-secure state, for Exception level n. The Exception levels are: **Bit[16]** Exception level 0. **Bit[17]** **RES0.** **Bit[18]** **RES0.** EXLEVEL_S[2] is never implemented. **Bit[19]** Exception level 3.
[15:12]	-	**RES0.**
[11]	TRCERR	Selects whether a system error exception must always be traced: **0** System error exception is traced only if the instruction or exception immediately before the system error exception is traced. **1** System error exception is always traced regardless of the value of ViewInst.
[10]	TRCRESET	Selects whether a reset exception must always be traced: **0** Reset exception is traced only if the instruction or exception immediately before the reset exception is traced. **1** Reset exception is always traced regardless of the value of ViewInst.
[9]	SSSTATUS	Indicates the current status of the start/stop logic: **0** Start/stop logic is in the stopped state. **1** Start/stop logic is in the started state.
[8]	-	**RES0.**

Bits	Name	Function
[7:0]	EVENT	An event selector.

2.1.14 Counter Reload Value Registers 0

The TRCCNTRLDVR0 defines the reload value for the reduced function counter.

Usage constraints

Only accepts writes when the trace unit is disabled.

The count value is only stable when TRCSTATR.PMSTABLE is 1.

If software uses counter0, then it must write to this register to set the counter reload value.

Configurations

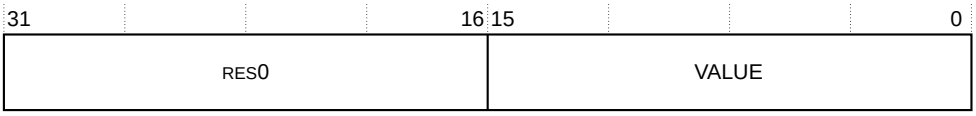
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-4: Derived resource register on page 24.

The following figure shows the TRCCNTRLDVR0 bit assignments.

Figure 2-12: TRCCNTRLDVR0 bit assignments



The following table shows the TRCCNTRLDVR0 bit assignments.

Table 2-22: TRCCNTRLDVR0 bit assignments

Bits	Value	Function
[31:16]	-	**RES0**.
[15:0]	VALUE	Defines the reload value for the counter. This value is loaded into the counter each time the reload event occurs.

2.1.15 ID Register 8-13

The TRCIDRn indicates information about the implemented trace stream.

Usage constraints

There are no usage constraints.

Configurations

These registers are available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-5: Implementation specific and identification registers on page 24.

The following figure shows the TRCIDR8 bit assignments.

Figure 2-13: TRCIDR8 bit assignments

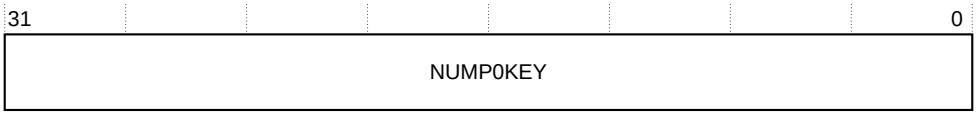
The following table shows the TRCIDR8 bit assignments.

Table 2-23: TRCIDR8 bit assignments

Bits	Name	Function
[31:0]	MAXSPEC	Indicates the maximum speculation depth of the instruction trace stream. This is the maximum number of P0 elements that have not been committed in the trace stream at any one time. **0x00000000** Maximum trace speculation depth is zero.

The following figure shows the TRCIDR9 bit assignments.

Figure 2-14: TRCIDR9 bit assignments



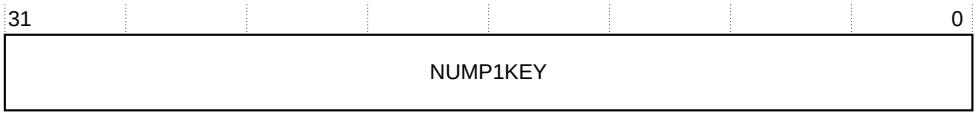
The following table shows the TRCIDR9 bit assignments.

Table 2-24: TRCIDR9 bit assignments

Bits	Name	Function
[31:0]	NUMP0KEY	Indicates the number of P0 right-hand keys that are used: **0x00000000** No P0 keys used in instruction trace.

The following figure shows the TRCIDR10 bit assignments.

Figure 2-15: TRCIDR10 bit assignments



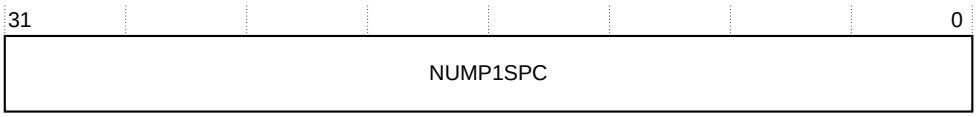
The following table shows the TRCIDR10 bit assignments.

Table 2-25: TRCIDR10 bit assignments

Bits	Name	Function
[31:0]	NUMP1KEY	Indicates the total number of P1 right-hand keys, including normal and special keys: **0x00000000** No P1 right-hand keys used in instruction trace.

The following figure shows the TRCIDR11 bit assignments.

Figure 2-16: TRCIDR11 bit assignments



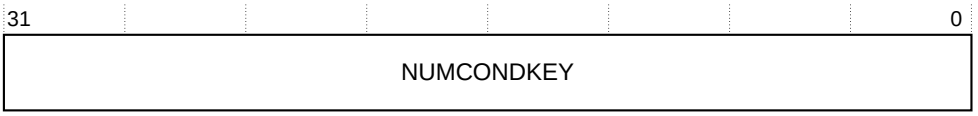
The following table shows the TRCIDR11 bit assignments.

Table 2-26: TRCIDR11 bit assignments

Bits	Name	Function
[31:0]	NUMP1SPC	Indicates the number of special P1 right-hand keys. **0x00000000** No special P1 right-hand keys used in any configuration.

The following figure shows the TRCIDR12 bit assignments.

Figure 2-17: TRCIDR12 bit assignments



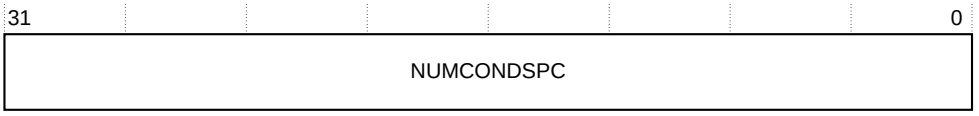
The following table shows the TRCIDR12 bit assignments.

Table 2-27: TRCIDR12 bit assignments

Bits	Name	Function
[31:0]	NUMCONDKEY	Indicates the total number of conditional instruction right-hand keys, including normal and special keys: **0x00000001** One conditional instruction right-hand key implemented.

The following figure shows the TRCIDR13 bit assignments.

Figure 2-18: TRCIDR13 bit assignments



The following table shows the TRCIDR13 bit assignments.

Table 2-28: TRCIDR13 bit assignments

Bits	Name	Function
[31:0]	NUMCONDSPC	This indicates the number of special conditional instruction right-hand keys. **0x00000000** No special conditional instruction right-hand keys implemented.

2.1.16 Implementation Specific Register 0

The TRCIMSPEC0 shows the presence of any **IMPLEMENTATION SPECIFIC** features, and enables any features that are provided.

Usage constraints

There are no usage constraints.

Configurations

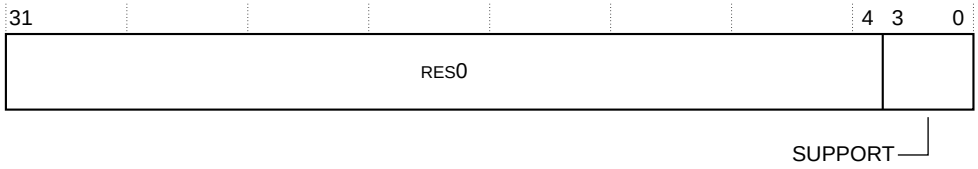
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-5: Implementation specific and identification registers on page 24.

The following figure shows the TRCIMSPEC0 bit assignments.

Figure 2-19: TRCIMSPEC0 bit assignments



The following table shows the TRCIMSPEC0 bit assignments.

Table 2-29: TRCIMSPEC0 bit assignments

Bits	Name	Function
[31:4]	-	**RES0.**
[3:0]	SUPPORT	Set to 0. No **IMPLEMENTATION SPECIFIC** extensions are supported.

2.1.17 ID Register 0

The TRCIDR0 indicates the tracing capabilities of the ETM-M33 instruction trace.

Usage constraints

There are no usage constraints.

Configurations

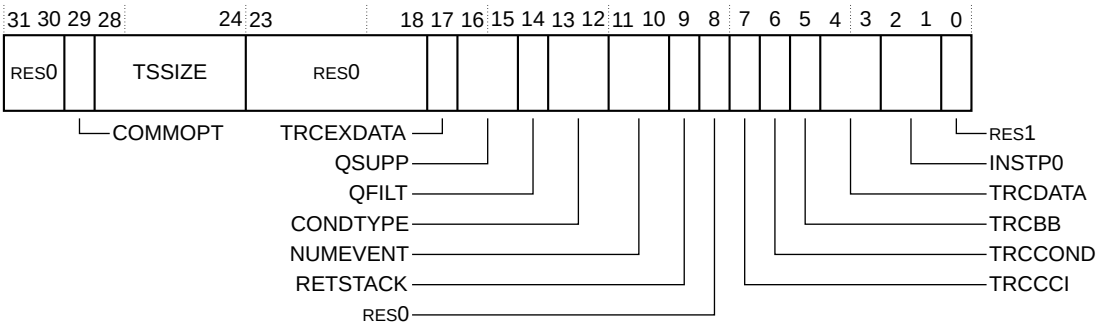
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-5: Implementation specific and identification registers on page 24.

The following figure shows the TRCIDR0 bit assignments.

Figure 2-20: TRCIDR0 bit assignments



The following table shows the TRCIDR0 bit assignments.

Table 2-30: TRCIDR0 bit assignments

Bits	Name	Function
[31:30]	-	**RES0.**
[29]	COMMOPT	Indicates the meaning of the commit field in some packets: **1** Commit mode 1.
[28:24]	TSSIZE	Global timestamp size: **0b01000** Maximum of 64-bit global timestamp implemented.
[23:18]	-	**RES0.**
[17]	TRCEXDATA	Indicates support for the tracing of data transfers for exceptions and exception returns: **0** TRCVDCTLR.TRCEXDATA is not implemented.
[16:15]	QSUPP	Indicates Q element support: **0b00** Q elements not supported.
[14]	QFILT	**RES0.**
[13:12]	CONDTYPE	Indicates how conditional results are traced: **0b00** The trace unit indicates only if a conditional instruction passes or fails its condition code check.
[11:10]	NUMEVENT	Number of events supported in the trace: **0b01** Two events supported.

Bits	Name	Function
[9]	RETSTACK	Return stack support: **1** Two entry return stack implemented.
[8]	-	**RES0.**
[7]	TRCCCI	Support for cycle counting in the instruction trace: **1** Cycle counting in the instruction trace is implemented.
[6]	TRCCOND	Support for conditional instruction tracing: **1** Conditional instruction tracing is implemented.
[5]	TRCBB	Support for branch broadcast tracing: **1** Branch broadcast tracing is implemented.
[4:3]	TRCDATA	Support for tracing of data: **0b00** Data tracing is not supported.
[2:1]	INSTP0	Support for tracing of load and store instructions as P0 elements: **0b00** Tracing of load and store instructions as P0 elements is not supported.
[0]	-	**RES1.**

Related information

Features on page 7

2.1.18 ID Register 1

The TRCIDR1 indicates the ETM-M33 architecture.

Usage constraints

There are no usage constraints.

Configurations

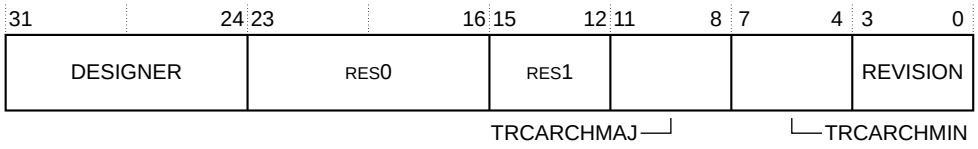
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-5: Implementation specific and identification registers on page 24.

The following figure shows the TRCIDR1 bit assignments.

Figure 2-21: TRCIDR1 bit assignments



The following table shows the TRCIDR1 bit assignments.

Table 2-31: TRCIDR1 bit assignments

Bits	Name	Function
[31:24]	DESIGNER	Indicates the designer of the trace unit: **0x41** Arm
[23:16]	-	**RES0**.
[15:12]	-	**RES1**.
[11:8]	TRCARCHMAJ	Major trace unit architecture version number: **0b0100** ETMv4.
[7:4]	TRCARCHMIN	Minor trace unit architecture version number: **0b0010** Minor revision 2.
[3:0]	REVISION	Implementation revision number: **0b0010** Implementation revision 2.

Related information

Features on page 7

2.1.19 ID Register 2

The TRCIDR2 indicates the maximum sizes of certain aspects of items in the trace.

Usage constraints

There are no usage constraints.

Configurations

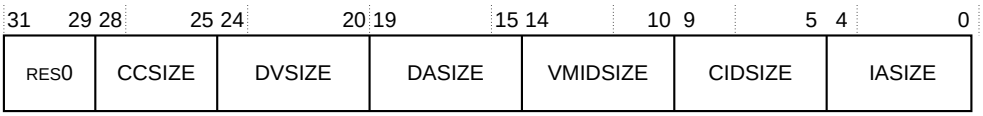
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-5: Implementation specific and identification registers on page 24.

The following figure shows the TRCIDR2 bit assignments.

Figure 2-22: TRCIDR2 bit assignments



The following table shows the TRCIDR2 bit assignments.

Table 2-32: TRCIDR2 bit assignments

Bits	Name	Function
[31:29]	-	**RES0**.
[28:25]	CCSIZE	Indicates the size of the cycle counter in bits minus 12: **0b0000** Cycle count is 12 bits in length.
[24:20]	DVSIZE	Data value size in bytes: **0b00000** Data value size not supported.
[19:15]	DASIZE	Data address size in bytes: **0b00000** Data address size not supported.
[14:10]	VMIDSIZE	Virtual Machine ID size: **0b00000** Virtual Machine ID tracing not implemented.
[9:5]	CIDSIZE	Context ID tracing: **0b00000** Context ID tracing not implemented.
[4:0]	IASIZE	Instruction address size: **0b00100** Maximum of 32-bit address size.

Related information

Features on page 7

2.1.20 ID Register 3

The TRCIDR3 indicates certain aspects of the ETM-M33 configuration.

Usage constraints

There are no usage constraints.

Configurations

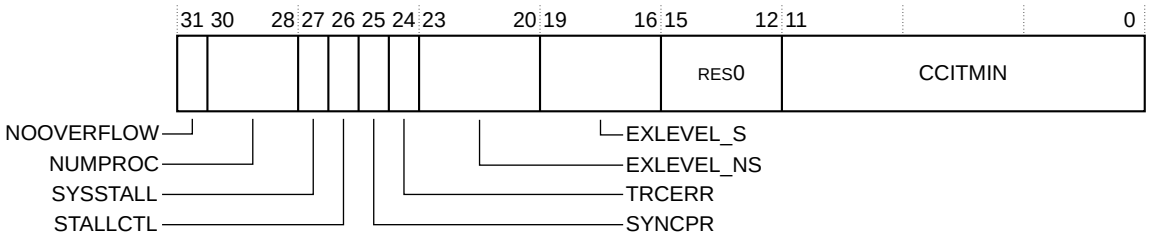
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-5: Implementation specific and identification registers on page 24.

The following figure shows the TRCIDR3 bit assignments.

Figure 2-23: TRCIDR3 bit assignments



The following table shows the TRCIDR3 bit assignments.

Table 2-33: TRCIDR3 bit assignments

Bits	Name	Function
[31]	NOOVERFLOW	Indicates whether NOOVERFLOW is implemented: **0** NOOVERFLOW is not implemented.
[30:28]	NUMPROC	Indicates the number of processors available for tracing. This is driven from the ETM-M33 NUMPROC input pin, reflecting system implementation: **0b000** The trace unit can trace one processor.
[27]	SYSSTALL	System support for stall control of the processor. This is driven from the ETM-M33 SYSSTALL input pin, reflecting the system implementation: **1** System supports stall control of the processor. This field is used with STALLCTL. Only when both SYSSTALL and STALLCTL are 1 does the system support stalling of the processor.
[26]	STALLCTL	Stall control support: **1** TRCSTALLCTLR is implemented.
[25]	SYNCPR	Indicates trace synchronization period support: **1** TRCSYNCPR is read-only for instruction trace only configuration. The trace synchronization period is fixed.
[24]	TRCERR	Indicates whether TRCVICTLR.TRCERR is implemented: **1** TRCERR is implemented.
[23:20]	EXLEVEL_NS	**RES0.**
[19:16]	EXLEVEL_S	Privilege levels implemented. One bit for each level. **0b1001** Privilege levels Thread and Handler are implemented.
[15:12]	-	**RES0.**
[11:0]	CCITMIN	Minimum value which can be programmed to TRCCCCTLR.THRESHOLD, defining the minimum cycle counting threshold. **0x4** Minimum of four instruction trace cycles.

Related information

Features on page 7

2.1.21 ID Register 4

The TRCIDR4 indicates the available ETM-M33 resources.

Usage constraints

There are no usage constraints.

Configurations

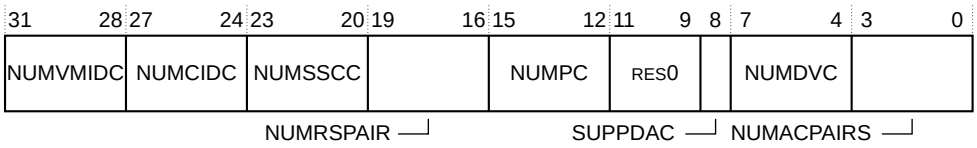
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-5: Implementation specific and identification registers on page 24.

The following figure shows the TRCIDR4 bit assignments.

Figure 2-24: TRCIDR4 bit assignments



The following table shows the TRCIDR4 bit assignments.

Table 2-34: TRCIDR4 bit assignments

Bits	Name	Function
[31:28]	NUMVMIDC	Number of *Virtual Machine ID* (VMID) comparators implemented: **0b0000** VMID comparators are not implemented.
[27:24]	NUMCIDC	Number of Context ID comparators implemented: **0b0000** Context ID comparators are not supported.
[23:20]	NUMSSCC	Number of single-shot comparator controls implemented: **0b0001** One single-shot comparator control is implemented.
[19:16]	NUMRSPAIR	Number of resource selection pairs implemented: **0b0001** Two resource selection pairs are implemented.
[15:12]	NUMPC	Number of processor comparator inputs implemented for the DWT: **0b0010** Two processor comparator inputs. **0b0100** Four processor comparator inputs.
[11:9]	-	**RES0.**

Bits	Name	Function
[8]	SUPPDAC	Data address comparisons implemented: **0** Data address comparisons are not supported.
[7:4]	NUMDVC	Number of data value comparators implemented: **0b0000** No data value comparators are implemented.
[3:0]	NUMACPAIRS	Number of address comparator pairs implemented: **0b0000** No address comparator pairs are implemented.

Related information

Features on page 7

2.1.22 ID Register 5

The TRCIDR5 indicates the available ETM-M33 resources.

Usage constraints

There are no usage constraints.

Configurations

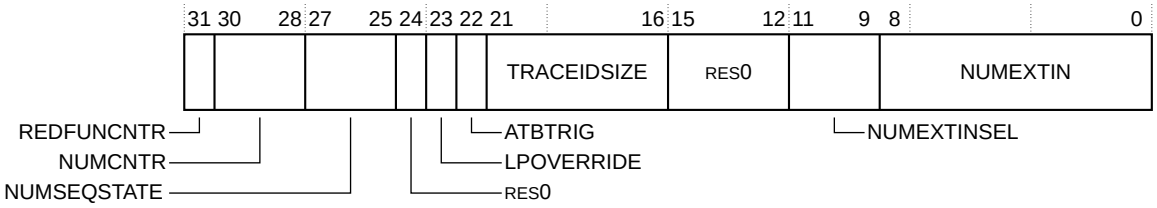
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-5: Implementation specific and identification registers on page 24.

The following figure shows the TRCIDR5 bit assignments.

Figure 2-25: TRCIDR5 bit assignments



The following table shows the TRCIDR5 bit assignments.

Table 2-35: TRCIDR5 bit assignments

Bits	Name	Function
[31]	REDFUNCNTR	Reduced Function Counter implemented: **1** Counter 0 is implemented as a Reduced Function Counter.

Bits	Name	Function
[30:28]	NUMCNTR	Number of counters implemented: **0b001** One counter implemented.
[27:25]	NUMSEQSTATE	Number of sequencer states implemented: **0b000** No sequencer states implemented.
[24]	-	**RES0.**
[23]	LPOVERRIDE	Low-power state override support: **1** Low-power state override support implemented.
[22]	ATBTRIG	ATB trigger support: **1** ATB trigger support implemented.
[21:16]	TRACEIDSIZE	Number of bits of trace ID: **0x07** Seven-bit trace ID implemented.
[15:12]	-	**RES0.**
[11:9]	NUMEXTINSEL	Number of external input selectors implemented: **0b000** No external input selectors are implemented.
[8:0]	NUMEXTIN	Number of external inputs implemented: **0x4** Four external inputs implemented.

Related information

Features on page 7

2.1.23 Resource Selection Registers 2-3

The TRCRSCTLRn controls the trace resources.

Usage constraints

There are no usage constraints.

Configurations

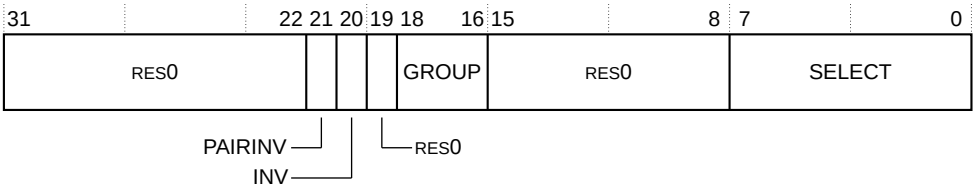
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-6: Resource selection register on page 25.

The following figure shows the TRCRSCTLRn bit assignments.

Figure 2-26: TRCRSCTLRn bit assignments



The following table shows the TRCRSCTLRn bit assignments.

Table 2-36: TRCRSCTLRn bit assignments

Bits	Name	Function
[31:22]	-	**RES0**.
[21]	PAIRINV	Inverts the result of a combined pair of resources. This bit is only implemented on the lower register for a pair of resource selectors.
[20]	INV	Inverts the selected resources: **0** Resource is not inverted. **1** Resource is inverted.
[19]	-	**RES0**.
[18:16]	GROUP	Selects a group of resources.
[15:8]	-	**RES0**.
[7:0]	SELECT	Selects one or more resources from the wanted group. One bit is provided per resource from the group.

2.1.24 Single-shot Comparator Control Register 0

The TRCSSCCR0 controls the single-shot comparator.

Usage constraints

There are no usage constraints.

Configurations

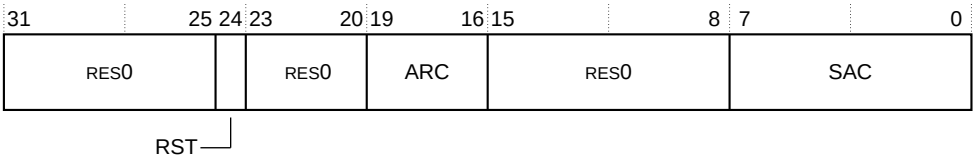
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-7: Single-shot comparator registers on page 25.

The following figure shows the TRCSSCCR0 bit assignments.

Figure 2-27: TRCSSCCR0 bit assignments



The following table shows the TRCSSCCR0 bit assignments.

Table 2-37: TRCSSCCR0 bit assignments

Bits	Name	Function
[31:25]	-	**RES0.**
[24]	RST	Enables the single-shot comparator resource to be reset when it occurs, to enable another comparator match to be detected: **1** Reset enabled. Multiple matches can occur.
[23:20]	-	**RES0.**
[19:16]	ARC	RAZ/WI.
[15:8]	-	**RES0.**
[7:0]	SAC	RAZ/WI.

2.1.25 Single-shot Comparator Status Register 0

The TRCSSCSR0 indicates the status of the single-shot comparators. TRCSSCSR0 is sensitive to instruction addresses.

Usage constraints

There are no usage constraints.

Configurations

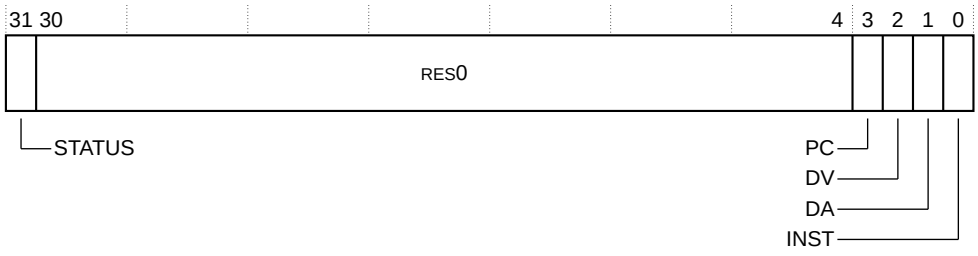
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-7: Single-shot comparator registers on page 25.

The following figure shows the TRCSSCSR0 bit assignments.

Figure 2-28: TRCSSCSR0 bit assignments



The following table shows the TRCSSCSR0 bit assignments.

Table 2-38: TRCSSCSR0 bit assignments

Bits	Name	Function
[31]	STATUS	Single-shot status. This indicates whether any of the selected comparators have matched: **0** Match has not occurred. **1** Match has occurred at least once. When programming ETM-M33, if TRCSSCCR0.RST is 0, the STATUS bit must be explicitly written to 0 to enable this single-shot comparator control.
[30:4]	-	**RES0.**
[3]	PC	Indicates that the Single-shot comparator is sensitive to processor comparator inputs: **1** Single-shot comparator is sensitive to processor comparator inputs.
[2]	DV	Data value comparator support: **0** Single-shot data value comparisons not supported.
[1]	DA	Data address comparator support: **0** Single-shot data address comparisons not supported.
[0]	INST	Instruction address comparator support: **0** Single-shot instruction address comparisons not supported.

2.1.26 Single-shot Processor Comparator Input Control Register

The TRCSSPCICR0 selects the processor comparator inputs for Single-shot control.

Usage constraints

Can only be written when the trace unit is disabled.

Configurations

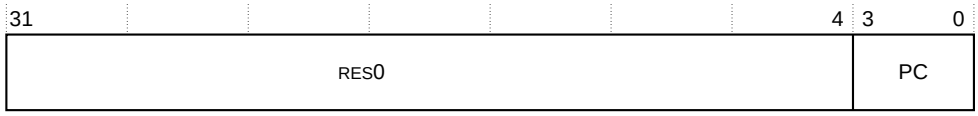
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-7: Single-shot comparator registers on page 25.

The following figure shows the TRCSSPCICR0 bit assignments.

Figure 2-29: TRCSSPCICR0 bit assignments



The following table shows the TRCSSPCICR0 bit assignments.

Table 2-39: TRCSSPCICR0 bit assignments

Bits	Name	Function
[31:4]	-	**RES0.**
[3:0]	PC	Selects one or more processor comparator inputs for Single-shot control. One bit is provided for each processor comparator input. The number of comparator inputs can be either two or four.

2.1.27 Power Down Control Register

The TRCPDCR request to the system power controller to keep ETM-M33 powered up.

Usage constraints

There are no usage constraints.

Configurations

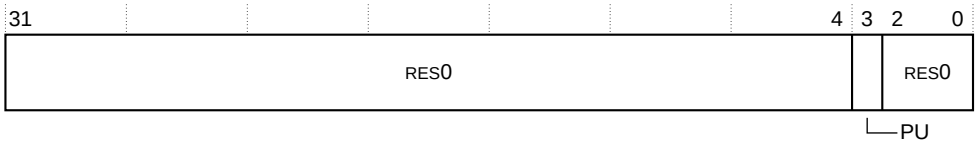
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-8: Power control registers on page 25.

The following figure shows the TRCPDCR bit assignments.

Figure 2-30: TRCPDCR bit assignments



The following table shows the TRCPDCR bit assignments.

Table 2-40: TRCPDCR bit assignments

Bits	Name	Function
[31:4]	-	**RES0.**

Bits	Name	Function
[3]	PU	Power up request, to request that power to ETM-M33 and access to the trace registers is maintained: **0** Power not requested. **1** Power requested. This bit is reset to 0 on a trace unit reset.
[2:0]	-	**RES0.**

2.1.28 Power Down Status Register

The TRCPDSR indicates the power down status of the ETM-M33.

Usage constraints

There are no usage constraints.

Configurations

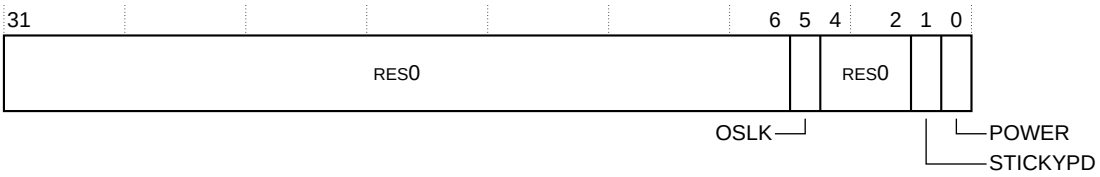
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-8: Power control registers on page 25.

The following figure shows the TRCPDSR bit assignments.

Figure 2-31: TRCPDSR bit assignments



The following table shows the TRCPDSR bit assignments.

Table 2-41: TRCPDSR bit assignments

Bits	Name	Function
[31:6]	-	**RES0.**
[5]	OSLK	**RES0.**
[4:2]	-	**RES0.**
[1]	STICKYPD	Sticky power down state. **0** Trace register power has not been removed since the TRCPDSR was last read. **1** Trace register power has been removed since the TRCPDSR was last read. This bit is set to 1 when power to the ETM-M33 registers is removed, to indicate that programming state has been lost. It is cleared after a read of the TRCPDSR.

Bits	Name	Function
[0]	POWER	Indicates ETM-M33 is powered up: **1** ETM-M33 is powered up. All registers are accessible. If a system implementation allows ETM-M33 to be powered down independently of the debug power domain, the system must ensure: • Accesses to ETM-M33 complete correctly. • Reads to this location return 0 to indicate that ETM-M33 is powered down.

2.1.29 Integration test registers

The ETM-M33 contains integration test registers. These can be used to access some of the ports that are useful in determining the system level trace topology by identifying the integration between specific components. Because the integration mode overrides the normal bus protocols, the ETM and ATB interconnect must be reset when any topology detection has been performed. Integration test registers are used to set the outputs and read the state of some of the signals.

To access the integration test registers, you must first set bit[0] of the 2.1.29.4 Integration Mode Control Register on page 57 to 1.

- You can use the write-only Integration test registers to set the outputs of some of the ETM-M33 signals. The following table shows the signals that can be controlled in this way.

Table 2-42: Output signals that the integration test registers can control

Signal	Register	Bits	Register description
AFREADYMI	TRCITIATBOUTR	[1]	2.1.29.3 Integration Instruction ATB Out Register on page 56
ATIDMI[6:0]	TRCITATBIDR	[6:0]	2.1.29.1 Integration ATB Identification Register on page 54
ATVALIDMI	TRCITIATBOUTR	[0]	2.1.29.3 Integration Instruction ATB Out Register on page 56

- You can use the read-only integration test registers to read the state of some of the ETM-M33 input signals. The following table shows the signals that can be read in this way.

Table 2-43: Input signals that the integration test registers can read

Signal	Register	Bits	Register description
AFVALIDMI	TRCITIATBINR	[1]	2.1.29.2 Integration Instruction ATB In Register on page 55
ATREADYMI	TRCITIATBINR	[0]	2.1.29.2 Integration Instruction ATB In Register on page 55

See the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* for more information about TRCITCTRL.

2.1.29.1 Integration ATB Identification Register

The TRCITATBIDR sets the state of output pins.

The output pins are listed in Table 2-44: TRCITATBIDR bit assignments on page 55.

Usage constraints

- Available when bit[0] of TRCITCTRL is set to 1.
- The value of the register sets the signals on the output pins when the register is written.

Configurations

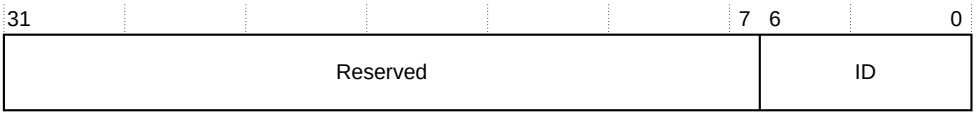
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22,Table 2-9: Integration test registers on page 25, and Table 2-42: Output signals that the integration test registers can control on page 54.

The following figure shows the TRCITATBIDR bit assignments.

Figure 2-32: TRCITATBIDR bit assignments



The following table shows the TRCITATBIDR bit assignments.

Table 2-44: TRCITATBIDR bit assignments

Bits	Name	Function
[31:7]	-	Reserved. Read undefined.
[6:0]	ID	Drives the ATIDMI[6:0] output pin.

2.1.29.2 Integration Instruction ATB In Register

The TRCITIATBINR reads the state of the input pins.

The input pins are listed in Table 2-45: TRCITIATBINR bit assignments on page 56.

Usage constraints

- Available when bit[0] of TRCITCTRL is set to 1.
- The values of the register bits depend on the signals on the input pins when the register is read.

Configurations

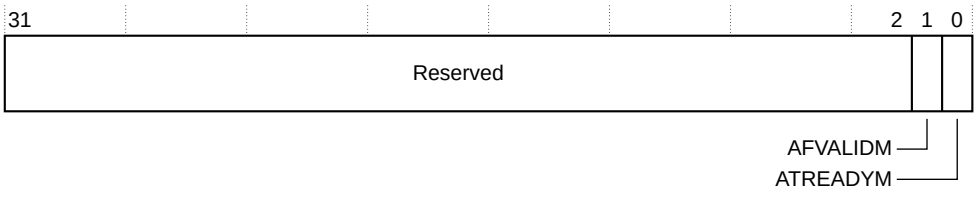
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22,Table 2-9: Integration test registers on page 25, and Table 2-42: Output signals that the integration test registers can control on page 54.

The following figure shows the TRCITIATBINR bit assignments.

Figure 2-33: TRCITIATBINR bit assignments



The following table shows the TRCITIATBINR bit assignments.

Table 2-45: TRCITIATBINR bit assignments

Bits	Name	Function
[31:2]	-	Reserved. Read undefined.
[1]	AFVALIDM	Returns the value of the AFVALIDMI input pin[4].
[0]	ATREADYM	Returns the value of the ATREADYMI input pin.

2.1.29.3 Integration Instruction ATB Out Register

The TRCITIATBOUTR sets the state of the output pins.

These output pins are listed inTable 2-46: TRCITIATBOUTR bit assignments on page 57.

Usage constraints

- Available when bit[0] of TRCITCTRL is set to 1.
- The value of the register sets the signals on the output pins when the register is written.

Configurations

Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22,Table 2-9: Integration test registers on page 25, and Table 2-42: Output signals that the integration test registers can control on page 54.

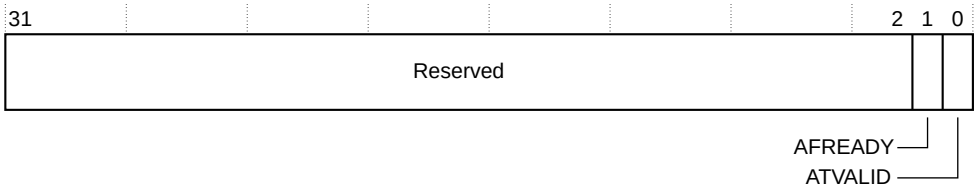
The following figure shows the TRCITIATBOUTR bit assignments.

[4] When an input pin is LOW, the corresponding register bit is 0.

When an input pin is HIGH, the corresponding register bit is 1.

The TRCITIATBINR bit values always correspond to the physical state of the input pins.

Figure 2-34: TRCITIATBOUTR bit assignments



The following table shows the TRCITIATBOUTR bit assignments.

Table 2-46: TRCITIATBOUTR bit assignments

Bits	Name	Function
[31:2]	-	Reserved. Read undefined.
[1]	AFREADY	Drives the AFREADYMI output pin.
[0]	ATVALID	Drives the ATVALIDMI output pin.

2.1.29.4 Integration Mode Control Register

The TRCITCTRL enables topology detection or integration testing, by putting ETM-M33 into integration mode.

Usage constraints

Arm recommends that you perform a debug reset after using integration mode.

Configurations

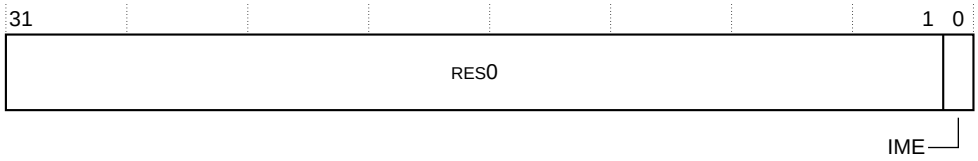
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-10: CoreSight management registers on page 26.

The following figure shows the TRCITCTRL bit assignments.

Figure 2-35: TRCITCTRL bit assignments



The following table shows the TRCITCTRL bit assignments.

Table 2-47: TRCITCTRL bit assignments

Bits	Name	Function
[31:1]	-	RES0.

Bits	Name	Function
[0]	IME	Integration mode enable: **0** ETM-M33 is not in integration mode. This is the reset value. **1** ETM-M33 is in integration mode.

2.1.30 Claim Tag Set Register

The TRCCLAIMSET sets bits in the claim tag and determines the number of claim tag bits implemented.

Usage constraints

There are no usage constraints.

Configurations

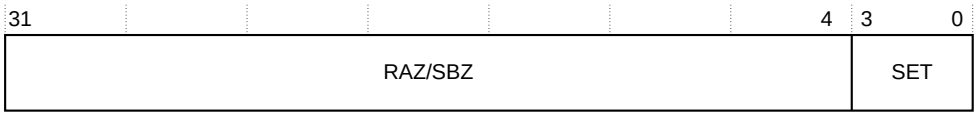
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-10: CoreSight management registers on page 26.

The following figure shows the TRCCLAIMSET bit assignments.

Figure 2-36: TRCCLAIMSET bit assignments



The following table shows the TRCCLAIMSET bit assignments.

Table 2-48: TRCCLAIMSET bit assignments

Bits	Name	Function
[31:4]	-	RAZ/SBZ.
[3:0]	SET	On reads, for each bit: **0** Claim tag bit is not implemented. **1** Claim tag bit is implemented. This value is returned for each of the claim bits 0-3, indicating 4 claim bits are implemented. On writes, for each bit: **0** Has no effect. **1** Sets the relevant bit of the claim tag.

2.1.31 Claim Tag Clear Register

The TRCCLAIMCLR clears bits in the claim tag and determines the current value of the claim tag.

Usage constraints

There are no usage constraints.

Configurations

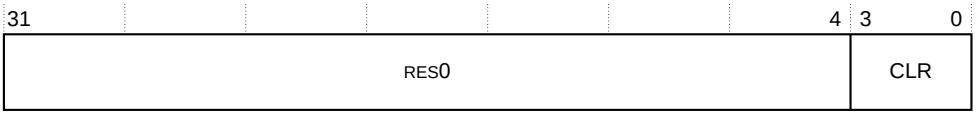
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-10: CoreSight management registers on page 26.

The following figure shows the TRCCLAIMCLR bit assignments.

Figure 2-37: TRCCLAIMCLR bit assignments



The following table shows the TRCCLAIMCLR bit assignments.

Table 2-49: TRCCLAIMCLR bit assignments

Bits	Name	Function
[31:4]	-	**RES0**.
[3:0]	CLR	On reads, for each bit: **0** Claim tag bit is not set. **1** Claim tag bit is set. On writes, for each bit: **0** Has no effect. **1** Clears the relevant bit of the claim tag.

2.1.32 Authentication Status Register

The TRCAUTHSTATUS indicates the current level of tracing permitted by the system.

Usage constraints

There are no usage constraints.

Configurations

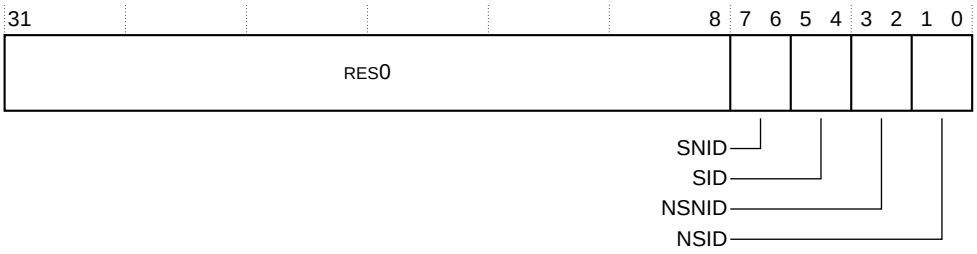
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-10: CoreSight management registers on page 26.

The following figure shows the TRCAUTHSTATUS bit assignments.

Figure 2-38: TRCAUTHSTATUS bit assignments



The following table shows the TRCAUTHSTATUS bit assignments.

Table 2-50: TRCAUTHSTATUS bit assignments

Bits	Name	Function
[31:8]	-	**RES0.**
[7:6]	SNID[5]	Secure Non-Invasive Debug: **0b10** Secure Non-Invasive Debug not implemented. **0b10** Secure Non-Invasive Debug implemented, but disabled. **0b11** Secure Non-Invasive Debug implemented and enabled.
[5:4]	SID	Secure Invasive Debug: **0b00** Secure Invasive Debug not implemented.
[3:2]	NSNID	Non-secure Non-Invasive Debug: **0b10** Non-secure Non-Invasive Debug implemented, but disabled. **0b11** Non-secure Non-Invasive Debug implemented and enabled.
[1:0]	NSID	Non-secure Invasive Debug: **0b00** Non-secure Invasive Debug not implemented.

2.1.33 Device Architecture Register

The TRCDEVARCH identifies ETM-M33 as an ETMv4.2 component.

Usage constraints

There are no usage constraints.

[5] SNID bitfield is RAZ when the processor is configured without the ARMv8-M Security Extension.

Configurations

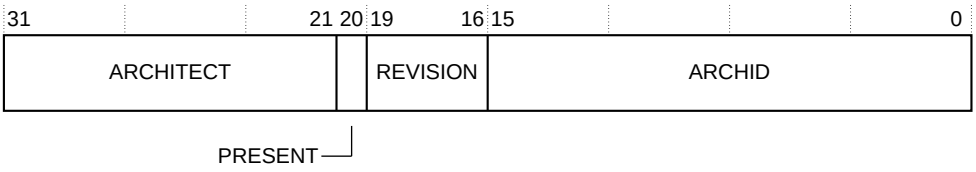
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-10: CoreSight management registers on page 26.

The following figure shows the TRCDEVARCH bit assignments.

Figure 2-39: TRCDEVARCH bit assignments



The following table shows the TRCDEVARCH bit assignments.

Table 2-51: TRCDEVARCH bit assignments

Bits	Name	Function
[31:21]	ARCHITECT	Defines the architect of the component: **0x23B** Arm.
[20]	PRESENT	Indicates the presence of this register: **0b1** Register is present.
[19:16]	REVISION	Architecture revision: **0b0010** Architecture revision 4.2.
[15:0]	ARCHID	Architecture ID: **0x4A13** ETMv4.2 component.

2.1.34 Device ID Register

The TRCDEVID is reserved.

Usage constraints

There are no usage constraints.

Configurations

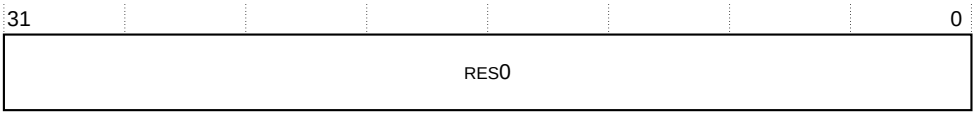
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-10: CoreSight management registers on page 26.

The following figure shows the TRCDEVID bit assignments.

Figure 2-40: TRCDEVID bit assignments



The following table shows the TRCDEVID bit assignments.

Table 2-52: TRCDEVID bit assignments

Bits	Name	Function
[31:0]	**RES0**	Reserved.

2.1.35 Device Type Register

The TRCDEVTYPE indicates the type of the component.

Usage constraints

There are no usage constraints.

Configurations

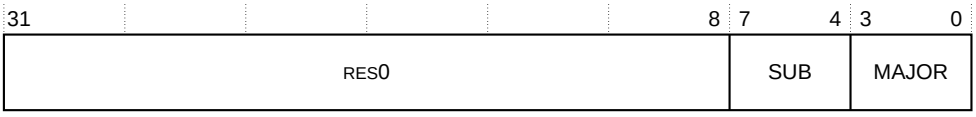
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-10: CoreSight management registers on page 26.

The following figure shows the TRCDEVTYPE bit assignments.

Figure 2-41: TRCDEVTYPE bit assignments



The following table shows the TRCDEVTYPE bit assignments.

Table 2-53: TRCDEVTYPE bit assignments

Bits	Name	Function
[31:8]	-	**RES0**.
[7:4]	SUB	The sub-type of the component: **0b0001** Processor trace.
[3:0]	MAJOR	The main type of the component: **0b0011** Trace source.

2.1.36 Peripheral Identification Registers

TRCPIDR[4-7, 0-3] provides the standard Peripheral ID required by all CoreSight components.

Usage constraints

Only bits[7:0] of each register are used. This means that TRCPIDR[4-7, 0-3] define a single 64-bit *Peripheral ID*, as the following figure shows.

Configurations

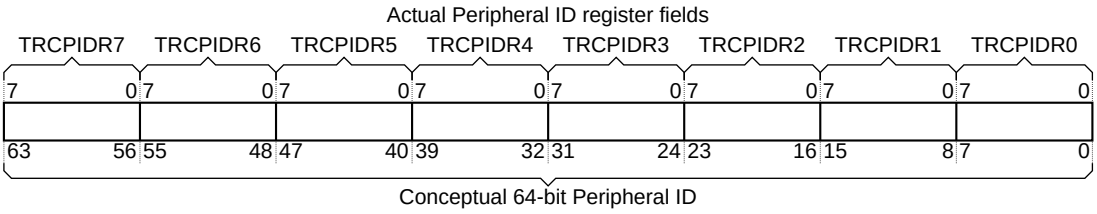
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-10: CoreSight management registers on page 26.

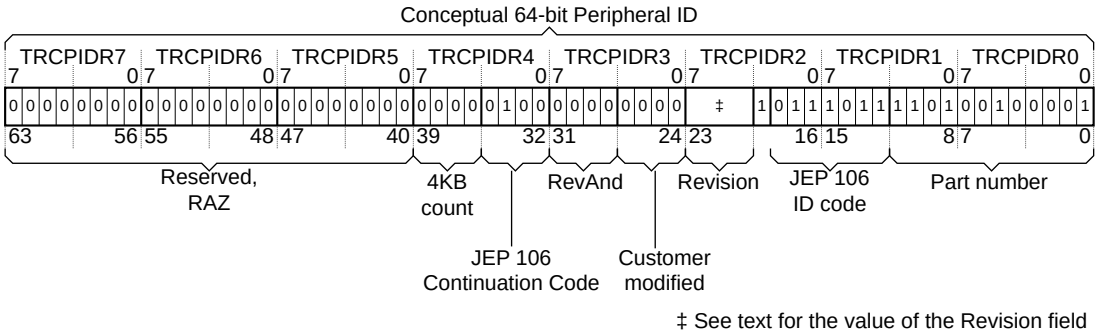
The following figure shows the mapping between TRCPIDR[4-7, 0-3] and the single 64-bit *Peripheral ID* value.

Figure 2-42: Mapping between TRCPIDR[4-7, 0-3] and the Peripheral ID value



The following figure shows the Peripheral ID bit assignments in the single conceptual Peripheral ID register.

Figure 2-43: Peripheral ID fields



The following table shows the values of the fields when reading this set of registers. The *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* gives more information about many of these fields.

The registers are listed in order of register name, from most significant (TRCPIDR7) to least significant (TRCPIDR0). This does not match the order of the register offsets.

Table 2-54: TRCPIDR[4-7, 0-3] bit assignments

Register	Register number	Register offset	Bits	Value	Description
TRCPIDR7	1015	0xFDC	[31:8]	-	**RES0**.
			[7:0]	0x00	**RES0**.
TRCPIDR6	1014	0xFD8	[31:8]	-	**RES0**.
			[7:0]	0x00	**RES0**.
TRCPIDR5	1013	0xFD4	[31:8]	-	**RES0**.
			[7:0]	0x00	**RES0**.
TRCPIDR4	1012	0xFD0	[31:8]	-	**RES0**.
			[7:4]	0x0	n, where 2^n is number of 4KB blocks used.
			[3:0]	0x4	JEP 106 continuation code.
TRCPIDR3	1019	0xFEC	[31:8]	-	**RES0**.
			[7:4]	0x0	RevAnd (at top level). Manufacturer revision number, ECOREVNUM[31:28].
			[3:0]	0x0	Customer Modified. 0x0 indicates from Arm.
TRCPIDR2	1018	0xFE8	[31:8]	-	**RES0**.
			[7:4]	See the Description column for more information.	Revision Number of Peripheral. This value is the same as the Implementation revision field of the TRCIDR1, see 2.1.18 ID Register 1 on page 42.
			[3]	0x1	Always 1. Indicates that a JEDEC assigned value is used.
			[2:0]	0x3	JEP 106 identity code [6:4].
TRCPIDR1	1017	0xFE4	[31:8]	-	**RES0**.
			[7:4]	0xB	JEP 106 identity code [3:0].
			[3:0]	0xD	Part Number[11:8].
TRCPIDR0	1016	0xFE0	[31:8]	-	**RES0**.
			[7:0]	0x21	Part Number [7:0].

2.1.37 Component Identification Registers

The TRCCIDR0-3 identifies ETM-M33 as a CoreSight component.

Usage constraints

Only bits[7:0] of each register are used. This means that TRCCIDR0-3 define a single 32-bit Component ID, as The following figure shows.

Configurations

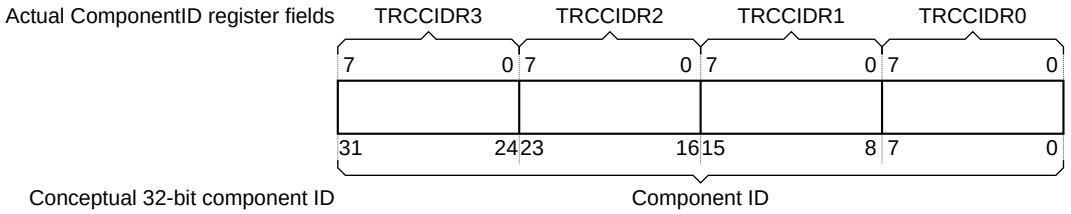
Available in all configurations.

Attributes

See the register summary in Table 2-1: ETM-M33 register summary on page 22 and Table 2-10: CoreSight management registers on page 26.

The following figure shows the mapping between TRCCIDR0-3 and the single 64-bit *Component ID* value.

Figure 2-44: Mapping between TRCCIDR0-3 and the Component ID value



The following table shows the Component ID bit assignments in the single conceptual Component ID register.

The registers are listed in order of register name, from most significant (TRCCIDR3) to least significant (TRCCIDR0). This does not match the order of the register offsets.

Table 2-55: TRCCIDR0-3 bit assignments

Register	Register number	Register offset	Bits	Value	Description
TRCCIDR3	0x3FF	0xFFC	[31:8]	-	**RES0**.
			[7:0]	0xB1	Component identifier, bits[31:24].
TRCCIDR2	0x3FE	0xFF8	[31:8]	-	**RES0**.
			[7:0]	0x05	Component identifier, bits[23:16].
TRCCIDR1	0x3FD	0xFF4	[31:8]	-	**RES0**.
			[7:4]	0x9	Debug component with CoreSight-compatible registers (component identifier, bits[15:12]).
			[3:0]	0x0	Component identifier, bits[11:8].
TRCCIDR0	0x3FC	0xFF0	[31:8]	-	**RES0**.
			[7:0]	0x0D	Component identifier, bits[7:0].

Proprietary Notice

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 the 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

This product is r0p2, which indicates the revision status of the product described in this manual, where:

r *(value)* Identifies the major revision of the product, for example, r1.
p *(value)* Identifies the minor revision or modification status of the product, for example, p2.

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
0002-06	1 August 2024	Non-Confidential	Fourth release for r0p2
0002-05	15 January 2023	Non-Confidential	Third release for r0p2
0002-01	25 November 2017	Non-Confidential	Second release for r0p2
0002-00	10 May 2017	Non-Confidential	First release for r0p2
0001-00	3 February 2017	Confidential	First release for r0p1

Issue	Date	Confidentiality	Change
0000-00	9 September 2016	Confidential	First release for r0p0.

The first table is for the first release. Then, each table compares the new issue of the manual with the last released issue of the manual. Issue numbers match the revision history in Document release information on page 68.

Table 2: Issue 0000-00

Change	Location
First Confidential release	-

Table 3: Differences between issue 0000-00 and issue 0001-00

Change	Location
First Confidential release for r0p1	-
Updated TRCIDR1.REVISION bit value to reflect product revision status	2.1.18 ID Register 1 on page 42
Corrected TRCIDR5.NUMEXTIN bit value to 4 in the function column	2.1.22 ID Register 5 on page 47

Table 4: Differences between issue 0001-00 and issue 0002-00

Change	Location
First Non-Confidential release for r0p2	-
Clarified the use of the integration test register in the ETM-M33	2.1.29 Integration test registers on page 54
Updated TRCIDR1.REVISION bit value to reflect product revision status	2.1.18 ID Register 1 on page 42

Table 5: Differences between issue 0002-00 and issue 0002-01

Change	Location
First Non-Confidential release for r0p3	-
Updated TRCIDR1.REVISION bit value to reflect product revision status.	2.1.18 ID Register 1 on page 42
Added TRCIDR10 to register summary table. Clarified TRCIDR3 reset value and removed footnote about Bits[30:27] being implementation dependent.	2.1.1 Register summary on page 21
Changed TRCSTALLCTLR.NOOVERFLOW to NOOVERFLOW in TRCIDR3 [31]bit function description.	2.1.20 ID Register 3 on page 44
Corrected conceptual 64-bit Peripheral ID JEP 106 ID code and part number bit field values.	Figure 2-43: Peripheral ID fields on page 63
Clarified register numbers in TRCPIDR[4-7, 0-3] bit assignments. Clarified TRCPIDR3 manufacturer revision number bitfield name, ECOREVNUM[31:28].	Table 2-54: TRCPIDR[4-7, 0-3] bit assignments on page 64

Table 6: Differences between issue 0002-01 and issue 0002-05

Change	Location
Third Non-Confidential release for r0p2	-
Changed topic title 'Additional reading' to 'Useful resources'	Useful resources

Table 7: Differences between issue 0002-05 and issue 0002-06

Change	Location
Fourth Non-Confidential release for r0p2	-
Changed book structure to move frontmatter content to backmatter	Across book
Added note for ATB triggers	1.2.3.10 Event tracing and triggers on page 18

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**.

Caution

We recommend the following. If you do not follow these recommendations your system might not work.

Warning

Your system requires the following. If you do not follow these requirements your system will not work.

Danger
You are at risk of causing permanent damage to your system or your equipment, or of harming yourself.

Note
This information is important and needs your attention.

Tip
This information might help you perform a task in an easier, better, or faster way.

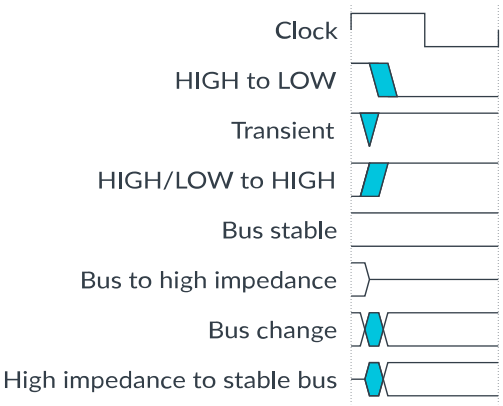
Remember
This information reminds you of something important relating to the current content.

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.

Access to Arm documents depends on their confidentiality:

- Non-Confidential documents are available at developer.arm.com/documentation. Each document link in the following tables goes to the online version of the document.
- Confidential documents are available to licensees only through the product package.

Arm product resources	Document ID	Confidentiality
Arm® CoreSight™ DAP-Lite2 Technical Reference Manual	100572	Non-Confidential
Arm® CoreSight™ SoC-400 Implementation Guide	DDI 0267	Confidential
Arm® CoreSight™ SoC-400 Technical Reference Manual	DDI 0480	Non-Confidential
Arm® CoreSight™ SoC-400 User Guide	DUI 0563	Confidential
Arm® Cortex®-M33 Processor Technical Reference Manual	100230	Non-Confidential
Arm® Cortex®-M33 Processor Integration and Implementation Manual	100323	Confidential

Arm architecture and specifications	Document ID	Confidentiality
AMBA® APB Protocol Version 2.0 Specification	IHI 0024	Non-Confidential
Arm® AMBA® 4 ATB Protocol Specification ATBv1.0 and ATBv1.1	IHI 0032	Non-Confidential
Arm® CoreSight™ Architecture Specification v3.0	IHI 0029	Non-Confidential
Arm® CoreSight™ Technology System Design Guide	DGI 0012	Non-Confidential
Arm® Embedded Trace Macrocell Architecture Specification ETMv4	IHI 0064	Non-Confidential