# Arm® CoreSight™ ETM-M55

**Revision: r0p2**

## Technical Reference Manual

**arm**

# Arm® CoreSight™ ETM-M55

## Technical Reference Manual

Copyright © 2019, 2020 Arm Limited or its affiliates. All rights reserved.

**Release Information**

### Document History

| Issue | Date | Confidentiality | Change |
|-------|------|-----------------|--------|
| 0000-02 | 07 June 2019 | Confidential | First beta release for r0p0 |
| 0000-04 | 20 December 2019 | Confidential | First limited access release for r0p0 |
| 0001-05 | 31 March 2020 | Non-Confidential | First early access release for r0p1 |
| 0002-01 | 17 July 2020 | Non-Confidential | First release for r0p2 |
| 0002-02 | 30 October 2020 | Non-Confidential | Second release for r0p2 |

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20349)

**Confidentiality Status**

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by Arm and the party that Arm delivered this document to.

Unrestricted Access is an Arm internal classification.

**Product Status**

The information in this document is Final, that is for a developed product.

**Web Address**

*developer.arm.com*

# Contents
# Arm® CoreSight™ ETM-M55 Technical Reference Manual

**Chapter 4**    **ETM-M55 register descriptions**

**Appendix A**    **Revisions**

# Preface

This preface introduces the *Arm® CoreSight™ ETM-M55 Technical Reference Manual*.

It contains the following:

## About this book

This book is for the CoreSight Embedded Trace Macrocell™ for the Cortex®-M55 processor. You implement ETM-M55 with the Cortex-M55 processor. In this manual, in general, any reference to the processor applies to the Cortex-M55 processor, as appropriate.

### Product revision status

The r*x*p*y* identifier indicates the revision status of the product described in this book, for example, r*1*p*2*, where:

r*x*    Identifies the major revision of the product, for example, r1.

p*y*    Identifies the minor revision or modification status of the product, for example, p2.

### 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®-M55 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®-M55 Processor Integration and Implementation Manual*.

### Using this book

This book is organized into the following chapters:

*Chapter 1 Introduction*
    This chapter describes ETM-M55.

*Chapter 2 Functional Description*
    This chapter describes the ETM-M55 functional features and operation.

*Chapter 3 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-M55 registers to control the macrocell.

*Chapter 4 ETM-M55 register descriptions*
    This chapter describes the ETM-M55 registers.

*Appendix A Revisions*
    This appendix describes the technical changes between released issues of this book.

#### 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.

#### Typographic conventions

*italic*
    Introduces special terminology, denotes cross-references, and citations.

**bold**
    Highlights interface elements, such as menu names. Denotes signal names. Also used for terms in descriptive lists, where appropriate.

`monospace`
    Denotes text that you can enter at the keyboard, such as commands, file and program names, and source code.

monospace
> Denotes a permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.

*monospace italic*
> Denotes arguments to monospace text where the argument is to be replaced by a specific value.

**monospace bold**
> Denotes language keywords when used outside example code.

<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
> Used in body text for a few terms that have specific technical meanings, that are defined in the *Arm® Glossary*. For example, IMPLEMENTATION DEFINED, IMPLEMENTATION SPECIFIC, UNKNOWN, and UNPREDICTABLE.

### 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.
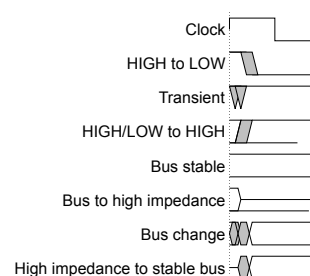


**Figure 1  Key to timing diagram conventions**

### 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.

## Additional reading

This book contains information that is specific to this product. See the following documents for other relevant information.

**Arm publications**

- *Arm® CoreSight™ DAP-Lite Technical Reference Manual* (DDI 0316)
- *Arm® CoreSight™ Architecture Specification v3.0* (IHI 0029)
- *AMBA® APB Protocol Version 2.0 Specification* (IHI 0033)
- *AMBA® 4 ATB Protocol Specification* (IHI 0032)
- *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* (ARM IHI 0064)
- *Arm®v8-M Architecture Reference Manual* (DDI 0553)
- *Arm® Cortex®-M55 Processor Technical Reference Manual* (101051)

The following confidential book is only available to licensees:

- *Arm® Cortex®-M55 Processor Integration and Implementation Manual* (101052)

**Other publications**

None.

# Feedback

## Feedback on this product

If you have any comments or suggestions about this product, contact your supplier and give:

- The product name.
- The product revision or version.
- An explanation with as much information as you can provide. Include symptoms and diagnostic procedures if appropriate.

## Feedback on content

If you have comments on content then send an e-mail to *errata@arm.com*. Give:

- The title *Arm CoreSight ETM-M55 Technical Reference Manual*.
- The number 101053_0002_02_en.
- If applicable, the page number(s) to which your comments refer.
- A concise explanation of your comments.

Arm also welcomes general suggestions for additions and improvements.

————— **Note** —————

Arm tests the PDF only in Adobe Acrobat and Acrobat Reader, and cannot guarantee the quality of the represented document when used with any other PDF reader.

————————————————

# Chapter 1
# Introduction

This chapter describes ETM-M55.

It contains the following sections:

## 1.1 CoreSight™ ETM-M55 features

ETM-M55 can provide non-intrusive program-flow trace for the Cortex-M55 processor. ETM-M55 generates information that trace software tools use to reconstruct the execution of an entire program or part of a program.

ETM-M55 implements instruction trace only. ETM-M55 is able to trace:

- All instructions, including condition code pass or fail
- Target addresses of indirect branch operations that have been taken
- Target addresses of direct branch operations that have been taken when TRCCONFIGR.BB is set to 1
- Exceptions
- Entry to debug state when Halting debug mode is enabled
- Cycle counts relating to instruction execution

ETM-M55 contains resource logic that enables you to control instruction trace. This resource logic includes one reduced function counter. For more information on the provision of a reduced function counter, see the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4*. Other resources available for ETM-M55 include *Data Watchpoint and Trace*(DWT) processor comparators and external inputs. You can specify the exact set of trigger and filter conditions that are required for a particular application.

ETM-M55 is a CoreSight component that provides instruction trace functionality.

For more information about CoreSight, see:
- *Arm® CoreSight™ Architecture Specification v3.0*
- *Arm® CoreSight™ System-on-Chip SoC-600 Technical Reference Manual*

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

### 1.1.1 The CoreSight™ debug environment

The CoreSight debug environment contains a software debugger that provides a user interface to ETM-M55. ETM-M55 is designed for instruction trace and it has a single 8-bit AMBA 4 ATB interface.

#### Software debugger

A software debugger provides a user interface to ETM-M55. You can use this interface to:

- Configure ETM-M55 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 registers and *Performance Monitoring Units* (PMUs).

ETM-M55 outputs its trace to the AMBA 4 ATB interface.

You can use the CoreSight infrastructure to design systems that provide the option to:
- Export the trace information through a trace port. An external *Trace Port Analyzer* (TPA) captures the trace information as shown in .
- 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 *Joint Test Action Group* (JTAG) or *Serial Wire* (SW) interface.

The software 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-M55 generates trace information that gives the software 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-M55 processor.

**Example CoreSight™ system with ETM-M55**

The following figure shows an example of how ETM-M55 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.
*   The base addresses of the components that are connected to the *External Private Peripheral Bus* (EPPB) on the Cortex-M55 processor.

The ETM-M55 trace interfaces are replicated to provide on-chip storage using the CoreSight ETB and output off-chip using the TPIU. Cross-triggering operates through the *Cross Trigger Interface* (CTI) and *Cross Trigger Matrix* (CTM) components.
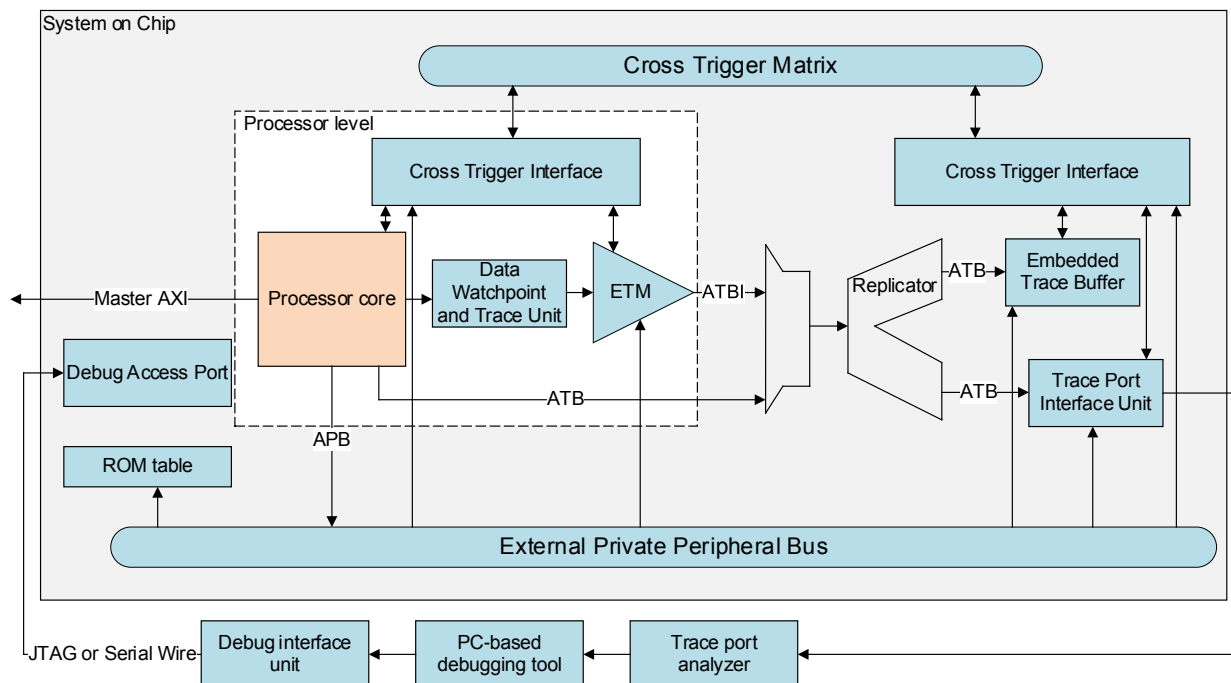


**Figure 1-1 Example CoreSight system with ETM-M55**

As an alternative to using an external computer to run a software debugger, the Cortex-M55 processor (or another processor on the *System-on-Chip* (SoC) can access ETM-M55 and an on-chip trace buffer to provide self-hosted debug and trace functionality.

## 1.2 Compliance

ETM-M55 is compatible with the CoreSight architecture and implements the architecture specification and protocols shown in the following table.

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

The following table shows the architecture specifications, protocols, and protocol specifications that the ETM-M55 is compatible with.

**Table 1-1  Architecture specifications and protocols**

| Component | Specification |
|---|---|
| ETM | ETM-M55 implements the ETM architecture version 4.5. For more information, see *Arm® Embedded Trace Macrocell Architecture Specification ETMv4*. |
| Interconnect architecture | ETM-M55 complies with the AMBA 4 APB and AMBA 4 ATB protocols. For more information, see the *AMBA® APB Protocol Version 2.0 Specification* and *AMBA® 4 ATB Protocol Specification*. |

## 1.3    Features

The ETM-M55 has 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 ETM version 4.5 features.
- Controlling tracing using triggering and filtering resources.

The following table shows the ETM-M55 features that are IMPLEMENTATION DEFINED, in terms of either:
- The number of times the feature is implemented.
- The size of the feature.

**Table 1-2  ETM-M55 features with IMPLEMENTATION DEFINED number of instances or size**

| Feature | ETM-M55 configuration | Description |
|---|---|---|
| Address comparators | 0 pairs | See bits[3:0] of the *4.16.5 TRCIDR4, ID Register 4* on page 4-61. |
| Instruction trace cycle counting minimum allowable threshold value | 4 | - |
| Data value comparators | 0 | See bits[7:4] of the *4.16.5 TRCIDR4, ID Register 4* on page 4-61. |
| Context ID comparators | 0 | See bits[27:24] *4.16.5 TRCIDR4, ID Register 4* on page 4-61. |
| Single-Shot comparator controls | 1 | The single-shot comparators are only sensitive to the processor comparator inputs. See bits [23:20] in *4.16.5 TRCIDR4, ID Register 4* on page 4-61. |
| Counters | 1 | This indicates reduced function counter implementation. See bits[30:28] of the *4.16.6 TRCIDR5, ID Register 5* on page 4-62. |
| Trace events supported | 2 | See bits[11:10] of the *4.16.1 TRCIDR0, ID Register 0* on page 4-57. |
| Cycle counter size | 12 bits | See bits[28:25] of the *4.16.3 TRCIDR2, ID Register 2* on page 4-59. |
| Sequencer | 0 | See bits[27:25] of the *4.16.6 TRCIDR5, ID Register 5* on page 4-62. |
| Processor comparator inputs | 2 or 4 | See bits[15:12] of the *4.16.5 TRCIDR4, ID Register 4* on page 4-61. |
| External inputs | 4+number of *Performance Monitor Unit* (PMU) events | See bits[8:0] of the *4.16.6 TRCIDR5, ID Register 5* on page 4-62. |
| External outputs | 2 | - |
| External input selectors | 4 | See bits[11:9] of the *4.16.6 TRCIDR5, ID Register 5* on page 4-62. |

**Table 1-2  ETM-M55 features with IMPLEMENTATION DEFINED number of instances or size (continued)**

| Feature | ETM-M55 configuration | Description |
|---|---|---|
| Resource selection pairs | 2 | See bits[19:16] of the *4.16.5 TRCIDR4, ID Register 4* on page 4-61. |
| Instruction trace port size | 8-bit | - |
| Instruction FIFO | 64 bytes with 8-bit output | Uses *Advanced Trace Bus* (ATB). |
| Claim tag bits | 4 | - |

The following table shows the optional features of the ETM architecture that ETM-M55 implements.

**Table 1-3  ETM-M55 implementation of optional features**

| Feature | Implemented | Description |
|---|---|---|
| Commit mode | No | - |
| Configurable FIFO | No | - |
| Trace Start/Stop block | Yes | - |
| Branch broadcast tracing support | Yes | See bit[5] of the *4.16.1 TRCIDR0, ID Register 0* on page 4-57. |
| Trace of conditional instructions | Yes | See bits[13:12] and bit[6] of the *4.16.1 TRCIDR0, ID Register 0* on page 4-57. |
| Data trace | No | - |
| System error trace | Yes | - |
| Load and store instruction data trace | No | - |
| Low-power override | Yes | See bit[23] of the *4.16.6 TRCIDR5, ID Register 5* on page 4-62. |
| Support for overflow avoidance | Yes | - |
| Cycle counting in instruction trace | Yes | See bit[7] of the *4.16.1 TRCIDR0, ID Register 0* on page 4-57. |
| Data address comparison | No | ETM-M55 does not implement data address comparison. |
| OS Lock mechanism | No | The Cortex-M55 processor does not implement OS Lock. |
| Secure non-invasive debug | Yes | The Cortex-M55 processor implements optional Security Extensions. |
| Context ID tracing | No | See bits[9:5] of the *4.16.3 TRCIDR2, ID Register 2* on page 4-59. |
| Trace output | Yes | Uses ATB. |
| Timestamp size | 64-bit | See bits[28:24] of the *4.16.1 TRCIDR0, ID Register 0* on page 4-57. |
| Memory mapped access to ETM-M55 registers | Yes | See the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* for more information about the access permissions behavior on register accesses for different states of ETM-Cortex-M55. |
| External debugger access to ETM-M55 registers | Yes | |
| System instruction access to ETM-M55 registers | No | |
| *Virtual Machine ID* (VMID) comparator support | No | See bits[31:28] of the *4.16.5 TRCIDR4, ID Register 4* on page 4-61. |
| Q-element filtering support | No | - |

**Table 1-3  ETM-M55 implementation of optional features (continued)**

| Feature | Implemented | Description |
|---|---|---|
| Q-element support | No | - |
| Reduced function counter | Yes | See bit [31] in *4.16.6 TRCIDR5, ID Register 5* on page 4-62. |
| Return stack support | Yes | - |
| Stall control support | Yes | - |
| Software lock | No | - |
| Synchronization period support | Yes | Read-only support is provided |
| ATB trigger support | Yes | See bit[22] of the *4.16.6 TRCIDR5, ID Register 5* on page 4-62. |

## 1.4      Interfaces and configurable options

ETM-M55 has four main interfaces.

### ETM-M55 interfaces

The following table shows the main interfaces in ETM-M55.

**Table 1-4  ETM-M55 interfaces**

| Interface | Description |
|---|---|
| Processor interface | This interface connects the M55 processor to ETM-M55. It has the following functions:<br>• Tracking execution information from the processor.<br>• Decoding the control signals.<br>• Passing on the decoded information to the internal interfaces. |
| AMBA 4 ATB interface | This is the instruction ATB interface. This interface reads single bytes of packet information from the instruction FIFO and the information is sent over the interface. |
| AMBA 4 APB interface | This is an interface to the APB that provides access to the programmable registers. |
| Production test interface | This interface contains the scan enable signal that is used in ETM-M55 production testing. |

### Configurable options

ETM-M55 has no configurable options.

## 1.5 Design process

ETM-M55 is delivered as synthesizable RTL.

Before it can be used in a product, it must go through the design process that is described in the following table.

**Table 1-5  ETM-M55 design process**

| Stage | Description |
|---|---|
| Implementation | For *Microcontroller Units* (MCUs), often a single design team integrates the processor and ETM-M55 before synthesizing the complete design. Alternatively, the team can synthesize the processor on its own or partially integrated, to produce a hard macrocell. Then, the same team or a separate team, integrates the hard macrocell into the design. |
| Integration | The integrator connects the implemented design into a *System on Chip* (SoC). This includes connecting it to a memory system and peripherals. |
| Programming | This is the last stage in the process. The system programmer develops the software required to configure and initialize ETM-M55, and tests the required application software. |

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 following:
> - Area.
> - Maximum frequency.
> - Features of the resulting design.

**Configuration inputs**

> The integrator configures some of the ETM-M55 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-M55 by programming particular values into registers. This affects the ETM-M55 behavior.

———— Note ————

- This manual refers to IMPLEMENTATION DEFINED features that are applicable to build configuration options. Any reference to:
  — An included feature means that the appropriate build and pin configuration options are selected.
  — An enabled feature means one that has also been configured by software.
- Each stage of the process:
  — Can be performed by a different party.
  — Can include implementation and integration choices that affect the ETM-M55 behavior and features.

## 1.6 Documentation

ETM-M55 documentation is as follows:

**Technical Reference Manual**

The *Technical Reference Manual* (TRM) describes the functionality and the effects of functional options on the ETM-M55 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-M55 is implemented and integrated.

**Integration and Implementation Manual**

For both the processor and ETM-M55, the *Arm® Cortex®-M55 Processor 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 *System on Chip* (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 documentation from your EDA tools vendor complements the IIM.

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

## 1.7    Product revisions

The following product revisions have been released.

r0p0        First release for Beta r0p0.

            First limited access release for r0p0.

r0p1        First early access release for r0p1.

r0p2        First release for r0p2.

            Second release for r0p2.

# Chapter 2
# Functional Description

This chapter describes the ETM-M55 functional features and operation.

It contains the following sections:

## 2.1 ETM-M55 functional blocks

ETM-M55 performs real-time instruction tracing based on the ETM architecture ETM version 4.5.

ETM-M55 is a CoreSight component and is an integral part of the Arm real-time debug solution, Arm Development Studio.

The following figure shows the main functional blocks of ETM-M55.
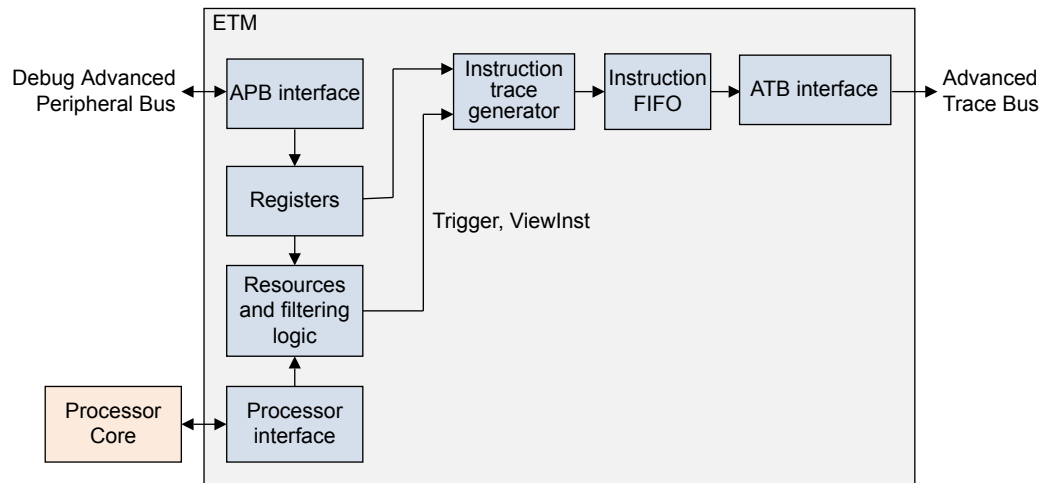


**Figure 2-1  ETM-M55 block diagram**

The following table describes the ETM-M55 functional blocks.

**Table 2-1  ETM-M55 functional blocks**

| Functional block | Description |
|---|---|
| Processor interface | The processor interface from ETM-M55 connects to the Cortex-M55 processor. It has the following functions:<br>• Tracking execution information from the processor.<br>• Decoding the control signals.<br>• Passing on the decoded information to the internal interfaces. |
| 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. |
| AMBA 4 APB interface | This is an interface to the APB that provides access to the programmable registers. |
| Instruction trace generator | This block generates the trace packets that are a compressed form of the instruction execution information provided by the Cortex-M55 processor. The trace packets are then passed to the instruction FIFO. |
| Instruction FIFO | This block buffers bursts of trace packets. There is one FIFO provided for the instruction trace stream. |
| AMBA 4 ATB interface | This is the instruction ATB interface. This interface reads single bytes of packet information from the instruction FIFO and the information is sent over the interface. |
| Global timestamping | ETM-M55 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 maximum processor clock frequency. |

## 2.2 External input and output connections

The following table shows the external input connections. These inputs can be selected using the external input selectors controlled by TRCEXTINSELR.

For more information, see *4.14 TRCEXTINSELR, External Input Select Register* on page 4-55.

**Table 2-2 External input connections**

| External input bits | Description |
|---|---|
| [3:0] | CTI |
| [3+N:4] where N is the number of PMU events | PMU events |

The following table shows the ETM-M55 external input and output bit connections to the *Cross Trigger Interface* (CTI).

**Table 2-3 CTI connections**

| ETM-M55 external input and output bits | CTI input |
|---|---|
| ETM external input bits [3:0] | CTI output bits [7:4] |
| ETM external output bits [1:0] | CTI input bits [5:4] |

The following table shows the ETM-M55 external input and output bit connections to the *Performance Monitoring Unit* (PMU).

**Table 2-4 PMU events**

| Event number | Event mnemonic | External input bit | Event name |
|---|---|---|---|
| 0x0000 | SW_INCR | 4 | Instruction architecturally executed, condition code check pass, software increment |
| 0x0001 | L1I_CACHE_REFILL | 5 | L1 instruction cache linefill |
| 0x0003 | L1D_CACHE_REFILL | 6 | L1 data cache linefill |
| 0x0004 | L1D_CACHE | 7 | L1 data cache access |
| 0x0006 | LD_RETIRED | 8 | Instruction architecturally executed, condition code check pass, load |
| 0x0007 | ST_RETIRED | 9 | Instruction architecturally executed, condition code check pass, store |
| 0x0008 | INST_RETIRED | 10 | Instruction architecturally executed. |
| 0x0009 | EXC_TAKEN | 11 | Exception taken. |
| 0x000A | EXC_RETURN | 12 | Instruction architecturally executed, condition code check pass, exception return. |
| 0x000C | PC_WRITE_RETIRED | 13 | Instruction architecturally executed, condition code check pass, software change of the PC. |
| 0x000D | BR_IMMED_RETIRED | 14 | Instruction architecturally executed, immediate branch. |

**Table 2-4  PMU events (continued)**

| Event number | Event mnemonic | External input bit | Event name |
|---|---|---|---|
| `0x000E` | BR_RETURN_RETIRED | 15 | Instruction architecturally executed, condition code check pass, procedure return. |
| `0x000F` | UNALIGNED_LDST_RETIRED | 16 | Instruction architecturally executed, condition code check pass, unaligned load or store. |
| `0x0011` | CPU_CYCLES | 17 | Cycle. |
| `0x0013` | MEM_ACCESS | 18 | Data memory access. |
| `0x0014` | L1I_CACHE | 19 | L1 instruction cache access. |
| `0x0015` | L1D_CACHE_WB | 20 | L1 data cache write-back |
| `0x0019` | BUS_ACCESS | 21 | Any beat access to the M-AXI read interface, M-AXI write interface and any access to P-AHB interface |
| `0x001A` | MEMORY_ERROR | 22 | ECC error for TCMs and caches. |
| `0x001D` | BUS_CYCLES | 23 | Counts the number of cycles on which the M-AXI interface is clocked. |
| `0x001E` | CHAIN | 24 | For an odd-numbered counter, increments when an overflow occurs on the preceding even-numbered counter on the same PE. |
| `0x0021` | BR_RETIRED | 25 | Instruction architecturally executed, branch. |
| `0x0022` | BR_MIS_PRED_RETIRED | 26 | Instruction architecturally executed, mispredicted branch. |
| `0x0023` | STALL_FRONTEND | 27 | If there are no instructions available from the fetch stage of the processor pipeline, the processor considers the front-end of the processor pipeline as being stalled. |
| `0x0024` | STALL_BACKEND | 28 | If there is an instruction available from the fetch stage of the pipeline but it cannot be accepted by the decode stage of the processor pipeline, the processor considers the back-end of the processor pipeline as being stalled. |
| `0x0036` | LL_CACHE_RD | 29 | L1 data cache read. For the Cortex-M55 processor, this event is the same as L1_CACHE_RD. |
| `0x0037` | LL_CACHE_MISS_RD | 30 | L1 data cache read miss. For the Cortex-M55 processor, this event is the same as L1D_CACHE_MISS_RD. |
| `0x0039` | L1D_CACHE_MISS_RD | 31 | L1 data cache read miss. For the Cortex-M55 processor, this event is the same as LL_CACHE_MISS_RD. |
| `0x003C` | STALL | 32 | No operation sent for execution. |

**Table 2-4  PMU events (continued)**

| Event number | Event mnemonic | External input bit | Event name |
|---|---|---|---|
| 0x0040 | L1D_CACHE_RD | 33 | L1 data cache read. For the Cortex-M55 processor, this event is the same as LL_CACHE_RD. |
| 0x0100 | LE_RETIRED | 34 | Loop end instruction architecturally executed, entry registered in the LO_BRANCH_INFO cache. |
| 0x0104 | BF_RETIRED | 35 | Tied LOW. |
| 0x0108 | LE_CANCEL | 36 | LO_BRANCH_INFO cache containing a valid loop entry cleared while not in the last iteration of the loop. |
| 0x0109 | BF_CANCEL | 37 | Tied LOW. |
| 0x0114 | SE_CALL_S | 38 | Call to secure function, resulting in security state change. |
| 0x0115 | SE_CALL_NS | 39 | Call to Non-secure function, resulting in security state change |
| 0x0200 | MVE_INST_RETIRED | 40 | *M-profile Vector Extension* (MVE) instruction architecturally executed |
| 0x0204 | MVE_FP_RETIRED | 41 | MVE floating-point instruction architecturally executed. |
| 0x0208 | MVE_FP_HP_RETIRED | 42 | MVE half-precision floating-point instruction architecturally executed |
| 0x020C | MVE_FP_SP_RETIRED | 43 | MVE single-precision floating-point instruction architecturally executed |
| 0x0214 | MVE_FP_MAC_RETIRED | 44 | MVE floating-point multiply or multiply accumulate instruction architecturally executed |
| 0x0224 | MVE_INT_RETIRED | 45 | MVE integer instruction architecturally executed |
| 0x0228 | MVE_INT_MAC_RETIRED | 46 | MVE integer multiply or multiply-accumulate instruction architecturally executed |
| 0x0238 | MVE_LDST_RETIRED | 47 | MVE load or store instruction architecturally executed |
| 0x023C | MVE_LD_RETIRED | 48 | MVE load instruction architecturally executed |
| 0x0240 | MVE_ST_RETIRED | 49 | MVE store instruction architecturally executed |
| 0x0244 | MVE_LDST_CONTIG_RETIRED | 50 | MVE contiguous load or store instruction architecturally executed |
| 0x0248 | MVE_LD_CONTIG_RETIRED | 51 | MVE contiguous load instruction architecturally executed |

**Table 2-4  PMU events (continued)**

| Event number | Event mnemonic | External input bit | Event name |
|---|---|---|---|
| 0x024C | MVE_ST_CONTIG_RETIRED | 52 | MVE contiguous store instruction architecturally executed |
| 0x0250 | MVE_LDST_NONCONTIG_RETIRED | 53 | MVE non-contiguous load or store instruction architecturally executed |
| 0x0254 | MVE_LD_NONCONTIG_RETIRED | 54 | MVE non-contiguous load instruction architecturally executed |
| 0x0258 | MVE_ST_NONCONTIG_RETIRED | 55 | MVE non-contiguous store instruction architecturally executed |
| 0x025C | MVE_LDST_MULTI_RETIRED | 56 | MVE memory instruction targeting multiple registers architecturally executed |
| 0x0260 | MVE_LD_MULTI_RETIRED | 57 | MVE memory load instruction targeting multiple registers architecturally executed |
| 0x0264 | MVE_ST_MULTI_RETIRED | 58 | MVE memory store instruction targeting multiple registers architecturally executed |
| 0x028C | MVE_LDST_UNALIGNED_RETIRED | 59 | MVE unaligned memory load or store instruction architecturally executed |
| 0x0290 | MVE_LD_UNALIGNED_RETIRED | 60 | MVE unaligned load instruction architecturally executed |
| 0x0294 | MVE_ST_UNALIGNED_RETIRED | 61 | MVE unaligned store instruction architecturally executed |
| 0x0298 | MVE_LDST_UNALIGNED_NONCONTIG_RETIRED | 62 | MVE unaligned non-contiguous load or store instruction architecturally executed |
| 0x02A0 | MVE_VREDUCE_RETIRED | 63 | MVE vector reduction instruction architecturally executed |
| 0x02A4 | MVE_VREDUCE_FP_RETIRED | 64 | MVE floating-point vector reduction instruction architecturally executed |
| 0x02A8 | MVE_VREDUCE_INT_RETIRED | 65 | MVE integer vector reduction instruction architecturally executed |
| 0x02B8 | MVE_PRED | 66 | Cycles where one or more predicated beats architecturally executed |
| 0x02CC | MVE_STALL | 67 | Stall cycles caused by an MVE instruction |
| 0x02CD | MVE_STALL_RESOURCE | 68 | Stall cycles caused by an MVE instruction because of resource conflicts |
| 0x02CE | MVE_STALL_RESOURCE_MEM | 69 | resource conflicts |
| 0x02CF | MVE_STALL_RESOURCE_FP | 70 | Stall cycles caused by an MVE instruction because of floating-point resource conflicts |
| 0x02D0 | MVE_STALL_RESOURCE_INT | 71 | Stall cycles caused by an MVE instruction because of integer resource conflicts |
| 0x02D3 | MVE_STALL_BREAK | 72 | Stall cycles caused by an MVE chain break |

**Table 2-4  PMU events (continued)**

| Event number | Event mnemonic | External input bit | Event name |
|---|---|---|---|
| 0x02D4 | MVE_STALL_DEPENDENCY | 73 | Stall cycles caused by MVE register dependency |
| 0x4007 | ITCM_ACCESS | 74 | *Instruction Tightly Coupled Memory* (ITCM) access |
| 0x4008 | DTCM_ACCESS | 75 | *Data Tightly Coupled Memory* (ITCM) access |
| 0x4010 | TRCEXTOUT0 | 76 | *Embedded Trace Macrocell* (ETM) external output 0 |
| 0x4011 | TRCEXTOUT1 | 77 | ETM external output 1 |
| 0x4012 | TRCEXTOUT2 | 78 | ETM external output 2 |
| 0x4013 | TRCEXTOUT3 | 79 | ETM external output 3 |
| 0x4018 | CTI_TRIGOUT4 | 80 | *Cross Trigger Interface* (CTI) output trigger 4 |
| 0x4019 | CTI_TRIGOUT5 | 81 | CTI output trigger 5 |
| 0x401A | CTI_TRIGOUT6 | 82 | CTI output trigger 6 |
| 0x401B | CTI_TRIGOUT7 | 83 | CTI output trigger 7 |
| 0xC000 | ECC_ERR | 84 | One or more *Error Correcting Code* (ECC) errors detected |
| 0xC001 | ECC_ERR_MBIT | 85 | One or more multi-bit ECC errors detected |
| 0xC010 | ECC_ERR_DCACHE | 86 | One or more ECC errors in the data cache |
| 0xC011 | ECC_ERR_ICACHE | 87 | One or more ECC errors in the instruction cache |
| 0xC012 | ECC_ERR_MBIT_DCACHE | 88 | One or more multi-bit ECC errors in the data cache |
| 0xC013 | ECC_ERR_MBIT_ICACHE | 89 | One or more multi-bit ECC errors in the instruction cache |
| 0xC020 | ECC_ERR_DTCM | 90 | One or more ECC errors in the DTCM |
| 0xC021 | ECC_ERR_ITCM | 91 | One or more ECC errors in the ITCM |
| 0xC022 | ECC_ERR_MBIT_DTCM | 92 | One or more multi-bit ECC errors in the DTCM |
| 0xC023 | ECC_ERR_MBIT_ITCM | 93 | One or more multi-bit ECC errors in the ITCM |
| 0xC100 | PF_LINEFILL | 94 | The prefetcher starts a linefill. |
| 0xC101 | PF_CANCEL | 95 | The prefetcher stops prefetching. |
| 0xC102 | PF_DROP_LINEFILL | 96 | A linefill triggered by the prefetcher has been dropped because of lack of buffering. |
| 0xC200 | NWAMODE_ENTER | 97 | No-write allocate mode entry |
| 0xC201 | NWAMODE | 98 | Write-Allocate store is not allocated into the data cache due to no-write-allocate mode |

**Table 2-4  PMU events (continued)**

| Event number | Event mnemonic | External input bit | Event name |
|---|---|---|---|
| `0xC300` | SAHB_ACCESS | 99 | Read or write access on the S-AHB interface to the TCM |
| `0xC301` | PAHB_ACCESS | 100 | Read or write access to the P-AHB write interface |
| `0xC302` | AXI_WRITE_ACCESS | 101 | Any beat access to M-AXI write interface. |
| `0xC303` | AXI_READ_ACCESS | 102 | Any beat access to M-AXI read interface. |
| `0xC400` | DOSTIMEOUT_DOUBLE | 103 | Denial of Service timeout has fired twice and caused buffers to drain to allow forward progress |
| `0xC401` | DOSTIMEOUT_TRIPLE | 104 | Denial of Service timeout has fired three times and blocked the LSU to force forward progress |

## 2.3 Operation

The ETM-M55 has IMPLEMENTATION DEFINED operations.

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

### 2.3.1 ETM-M55 registers

There are two groups of ETM-M55 registers:

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

### 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*.

### 2.3.3 Parallel instruction execution

The Cortex-M55 processor supports parallel instruction execution. This means that ETM-M55 is capable of tracing two instructions per cycle.

If **ViewInst** is active in a cycle when two instructions are executed, then both instructions are traced.

### 2.3.4 Trace features

The ETM-M55 implements the following optional ETMv4.5 trace features:

- Cycle-counting
- Timestamping
- Branch broadcasting
- Conditional instruction tracing

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

### 2.3.5 Packet formats

This section describes the packet formats that the ETM-M55 instruction trace interface supports.

The ETM-M55 instruction trace interface does not support the following trace packet types:
- Speculation resolution packets are not supported.
- Q instruction trace packets are not supported.

ETM-M55 supports conditional tracing, but does not trace APSR condition flag values. See the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* for the trace packet format descriptions.

### 2.3.6 Resource selection

The ETM-M55 uses event selectors to control resources.

The ETM-M55 has the following resources:

- Reduced function counter
- *Data Watchpoint and Trace* (DWT) processor comparators
- Single shot comparator
- External inputs

The ETM-M55 generates the following events:

- Trace events, triggers, and markers in the trace stream
- Timestamp event
- ViewInst event

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-M55 are located in the following registers:

- Event Control 0 Register, TRCEVENTCTL0R
- Global Timestamp Control Register, TRCTSCTLR
- ViewInst Main Control Register, TRCVICTLR

The ETM-M55 provides:

- A fixed resource selector pair, registers TRCRSCTLR0 and TRCRSCTLR1 with static values of 0 = FALSE and 1 = TRUE, respectively
- A 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 2-5  Instruction trace resource selection**

| Group | Select | Resource |
|---|---|---|
| 0b0000 | Bits 0-3 | External input selectors 0-3<br><br>When select bit N is set, then the resource selector is sensitive to external input selector N. |
| 0b0001 | Bits 0-3 | When select bit N is set, the resource selector is sensitive to processor comparator input N. |
| 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 *Advanced Trace Bus* (ATB) trigger. This example uses the user-configurable resource selector 2.

The DWT unit controls the processor comparator inputs.



| | |
|---|---|
| TRCSSPCICR0 <0x000008> | Select processor comparator input 3 for single-shot comparator 0 |
| TRCRSCTL2 <0x30000> | Resource selector selects resource selector 2 |
| TRCEVENTCTL0R <0x02> | Event selector sensitive to single resource 2 |
| TRCEVENTCTL1R <0x801> | Event 0 generates event element and ATB trigger |
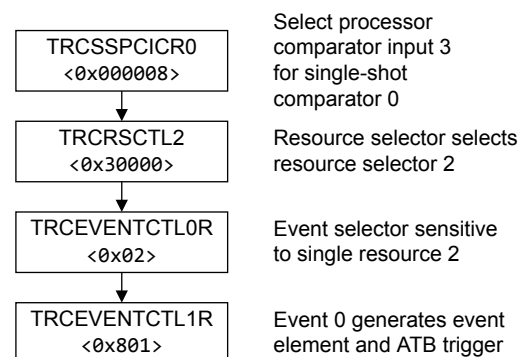
**Figure 2-2  Trigger event resource selection**

The ETM-M55 single-shot and start-stop logic might not reliably trigger when DWT comparators are programmed for comparisons other than instruction address comparison.

### 2.3.7 Trace flush behavior

Events that ETM-M55 observes can be confirmed to have reached the trace bus output by using the ATB flush protocol.

The ETM-M55 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-M55.

### 2.3.8 Low-power state behavior

When the processor enters a low-power state, there is a delay before the resources in the ETM-M55 become inactive.

This permits the last instruction executed to trigger a comparator or update the counter, and the resulting event packet to be inserted in the specified trace stream. This event packet is presented on the trace bus before the ETM-M55 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-M55 enters a low-power state, but is traced when the processor leaves the low-power state, if the ETM-M55 logic is not reset before this can occur.

This low-power behavior can be disabled using TRCEVENTCTL1R.LPOVERRIDE bit. For more information on TRCEVENTCTL1R, see *4.6 TRCEVENTCTL1R, Event Control 1 Register on page 4-45*. In this case, the ETM-M55 resources remain active.

### 2.3.9 Cycle counter

The ETM-M55 uses a 12-bit cycle counter.

The cycle counter does not count when non-invasive debug is disabled or when ETM-M55 is in a low-power state.

### 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 because the rate at which the events enter the FIFO is the same as the rate at which events leave the FIFO. Therefore, the instruction FIFO overflows.

When used with the optimized Cortex-M55 TPIU, *Advanced Trace Bus* (ATB) triggers must not be enabled, TRCEVENTCTL1R.ATB must be set to 0. For more information on TRCEVENTCTL1R, see *4.6 TRCEVENTCTL1R, Event Control 1 Register on page 4-45*.

# Chapter 3
# 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-M55 registers to control the macrocell.

It contains the following section:

## 3.1 Modes of operation and execution

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

### 3.1.1 Controlling ETM-M55 programming

When programming the ETM-M55 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 Programming Control Register, TRCPRGCTLR.
- The Status Register, TRCSTATR, to indicate the ETM-M55 status.

The following figure shows the procedure to use.

```
                    ( Start )
                        |
             ┌──────────────────────┐
             │  Set TRCPRGCTLR.EN    │
             │     bit to 0b0        │
             └──────────────────────┘
                        |
             ┌──────────────────────┐ ◄─────────┐
             │    Read TRCSTATR      │           │ No
             └──────────────────────┘           │
                        |                        │
                  ╱ Is TRCSTATR.IDLE ╲ ──────────┘
                  ╲     0b1?         ╱
                        |
                      Yes
                        |
             ┌──────────────────────┐
             │  Program all trace    │
             │  registers required   │
             └──────────────────────┘
                        |
             ┌──────────────────────┐
             │  Set TRCPRGCTLR.EN    │
             │     bit to 0b1        │
             └──────────────────────┘
                        |
                        ◄──────────────┐
                        |               │ No
                  ╱ Is TRCSTATR.IDLE ╲ ─┘
                  ╲     0b0?         ╱
                        |
                      Yes
                        |
                    ( End )
```
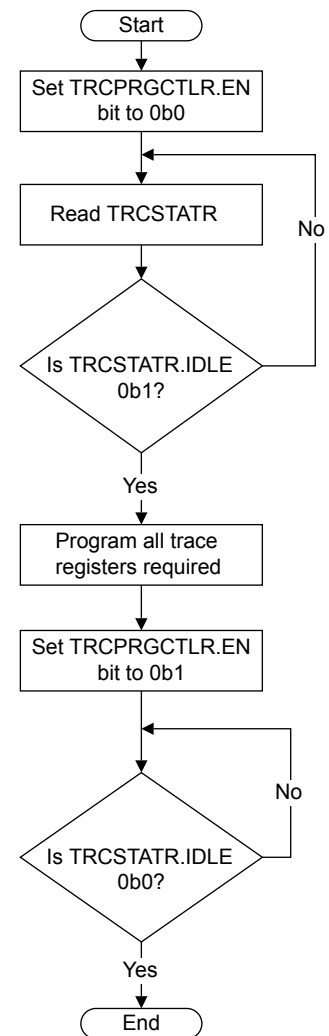
**Figure 3-1  Programming ETM registers**

The Cortex-M55 processor does not have to be in debug state while you program the ETM-M55 registers.

To access the ETM registers, use the external *Advanced Peripheral Bus* (APB) interface. This provides a direct method of programming the ETM.

# Chapter 4
# ETM-M55 register descriptions

This chapter describes the ETM-M55 registers.

These registers can be categorized into:
- General control and ID registers.
- Trace filtering control register.
- Derived resource register.
- Implementation specific and identification registers.
- Resource selection register.
- Single-shot comparator registers.
- Power control registers.
- Integration test registers.
- CoreSight management registers.

It contains the following sections:

## 4.1 Register summary

All ETM-M55 registers are 32 bits wide.

──────── **Note** ────────

Registers not listed in the following table and not described in this document are not implemented. Reading a non-implemented register address returns zero. Writing to a non-implemented register address has no effect.

────────────────

The table below:
- Lists the ETM-M55 registers in numerical order and describes each register.
- Includes the following additional information about each register:
  — 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*.
  — The register access type.
  — Additional information about the implementation of the register, where appropriate.

**Table 4-1  ETM-M55 register summary**

| Register number | Base offset | Name | Type | Reset value | Description |
|---|---|---|---|---|---|
| 1 | 0x004 | TRCPRGCTLR | RW | 0x00000000 | *4.2 TRCPRGCTLR, Programming Control Register on page 4-40* |
| 3 | 0x00C | TRCSTATR | RO | UNKNOWN | *4.3 TRCSTATR, Status Register on page 4-41* |
| 4 | 0x010 | TRCCONFIGR | RW | UNKNOWN | *4.4 TRCCONFIGR, Trace Configuration Register on page 4-42* |
| 6 | 0x018 | TRCAUXCTLR | RW | 0x00000000 | Auxiliary Control Register. |
| 8 | 0x020 | TRCEVENTCTL0R | RW | UNKNOWN | *4.5 TRCEVENTCTL0R, Event Control 0 Register on page 4-44* |
| 9 | 0x024 | TRCEVENTCTL1R | RW | UNKNOWN | *4.6 TRCEVENTCTL1R, Event Control 1 Register on page 4-45* |
| 11 | 0x02C | TRCSTALLCTLR | RW | UNKNOWN | *4.7 TRCSTALLCTLR, Stall Control Register on page 4-46* |
| 12 | 0x030 | TRCTSCTLR | RW | UNKNOWN | *4.8 TRCTSCTLR, Global Timestamp Control Register on page 4-47* |
| 13 | 0x034 | TRCSYNCPR | RO | 0x0000000A | *4.9 TRCSYNCPR, Synchronization Period Register on page 4-48* |
| 14 | 0x038 | TRCCCCTLR | RW | UNKNOWN | *4.10 TRCCCCTLR, Cycle Count Control Register on page 4-49* |
| 16 | 0x040 | TRCTRACEIDR | RW | UNKNOWN | *4.11 TRCTRACEIDR, Trace ID Register on page 4-50* |
| 32 | 0x080 | TRCVICTLR | RW | UNKNOWN | *4.12 TRCVICTLR, ViewInst Main Control Register on page 4-51* |
| 35 | 0x08C | TRCVIPCSSCTLR | RW | UNKNOWN | *4.12 TRCVICTLR, ViewInst Main Control Register on page 4-51* |
| 72 | 0x120 | TRCEXTINSELR | RW | UNKNOWN | *4.14 TRCEXTINSELR, External Input Select Register on page 4-55* |

**Table 4-1  ETM-M55 register summary (continued)**

| Register number | Base offset | Name | Type | Reset value | Description |
|---|---|---|---|---|---|
| 80 | 0x140 | TRCCNTRLDVR0 | RW | UNKNOWN | *4.15 TRCCNTRLDVR0, Counter Reload Value Register 0 on page 4-56* |
| 96 | 0x180 | TRCIDR8 | RO | 0x00000000 | *4.16.9 TRCIDR8, ID Register 8 on page 4-64* |
| 97 | 0x184 | TRCIDR9 | RO | 0x00000000 | *4.16.10 TRCIDR9, ID Register 9 on page 4-64* |
| 98 | 0x188 | TRCIDR10 | RO | 0x00000000 | *4.16.11 TRCIDR10, ID Register 10 on page 4-65* |
| 99 | 0x18C | TRCIDR11 | RO | 0x00000000 | *4.16.12 TRCIDR11, ID Register 11 on page 4-65* |
| 100 | 0x190 | TRCIDR12 | RO | 0x00000001 | *4.16.13 TRCIDR12, ID Register 12 on page 4-65* |
| 101 | 0x194 | TRCIDR13 | RO | 0x00000000 | *4.16.14 TRCIDR13, ID Register 13 on page 4-66* |
| 112 | 0x1C0 | TRCIMSPEC0 | RW | 0x00000000 | Implementation specific Register 0. There are not IMPLEMENTATION SPECIFIC extensions supported, and this register is not implemented. |
| 120 | 0x1E0 | TRCIDR0 | RO | 0x280006E1 | *4.16.1 TRCIDR0, ID Register 0 on page 4-57* |
| 121 | 0x1E4 | TRCIDR1 | RO | 0x4100F453 | *4.16.2 TRCIDR1, ID Register 1 on page 4-58* |
| 122 | 0x1E8 | TRCIDR2 | RO | 0x00000004 | *4.16.3 TRCIDR2, ID Register 2 on page 4-59* |
| 123 | 0x1EC | TRCIDR3 | RO | 0x0F090004 | *4.16.4 TRCIDR3, ID Register 3 on page 4-60* |
| 124 | 0x1F0 | TRCIDR4 | RO | -[a] | *4.16.5 TRCIDR4, ID Register 4 on page 4-61* |
| 125 | 0x1F4 | TRCIDR5 | RO | 0x90C70004 | *4.16.6 TRCIDR5, ID Register 5 on page 4-62* |
| 126 | 0x1F8 | TRCIDR6 | - | UNKNOWN | Reserved, RES0 |
| 127 | 0x1FC | TRCIDR7 | - | UNKNOWN | Reserved, RES0 |
| 130-131 | 0x208-0x20C | TRCRSCTLR2-3 | RW | UNKNOWN | *4.17 TRCRSCTLRn, Resource Selection Registers 2-3 on page 4-67* |
| 160 | 0x280 | TRCSSCCR0 | RW | UNKNOWN | *4.18 TRCSSCCR0, Single-shot Comparator Control Register 0 on page 4-69* |
| 168 | 0x2A0 | TRCSSCSR0 | RW | UNKNOWN | *4.19 TRCSSCSR0, Single-shot Comparator Status Register 0 on page 4-70* |
| 176 | 0x2C0 | TRCSSPCICR0 | RW | UNKNOWN | *4.20 TRCSSPCICR0, Single-shot Processor Comparator Input Control Register 0 on page 4-72* |
| 196 | 0x310 | TRCPDCR | RW | 0x00000000 | *4.21 TRCPDCR, Power Down Control Register on page 4-73* |
| 197 | 0x314 | TRCPDSR | RO | 0x00000003 | *4.22 TRCPDSR, Power Down Status Register on page 4-74* |
| 953 | 0xEE4 | TRCITATBIDR | WO | UNKNOWN | *4.23.1 TRCITATBIDR, Integration ATB Identification Register on page 4-76* |
| 955 | 0xEEC | TRCITIDATAR | WO | UNKNOWN | *4.23.2 TRCITIDATAR, Integration Data Register on page 4-77* |

[a]
0x00114000    For 4 comparator configuration
0x00112000    For 2 comparator configuration

**Table 4-1  ETM-M55 register summary (continued)**

| Register number | Base offset | Name | Type | Reset value | Description |
|---|---|---|---|---|---|
| 957 | 0xEF4 | TRCITIATBINR | RO | UNKNOWN | *4.23.3 TRCITIATBINR, Integration Instruction ATB In Register* on page 4-77 |
| 959 | 0xEFC | TRCITIATBOUTR | WO | UNKNOWN | *4.23.4 TRCITIATBOUTR, Integration Instruction ATB Out Register* on page 4-78 |
| 960 | 0xF00 | TRCITCTRL | RW | 0x00000000 | *4.31 TRCCIDR0-3, Component Identification Registers* on page 4-88 *4.23.5 TRCITCTRL, Integration Mode Control Register* on page 4-79 |
| 1000 | 0xFA0 | TRCCLAIMSET | RW | 0x0000000F | *4.24 TRCCLAIMSET, Claim Tag Set Register* on page 4-80 |
| 1001 | 0xFA4 | TRCCLAIMCLR | RW | 0x00000000 | *4.25 TRCCLAIMCLR, Claim Tag Clear Register* on page 4-81 |
| 1006 | 0xFB8 | TRCAUTHSTATUS | RO | UNKNOWN | *4.26 TRCAUTHSTATUS, Authentication Status Register* on page 4-82 |
| 1007 | 0xFBC | TRCDEVARCH | RO | 0x47754A13 | *4.27 TRCDEVARCH, Device Architecture Register* on page 4-83 |
| 1010 | 0xFC8 | TRCDEVID | RO | 0x00000000 | *4.28 TRCDEVID, Device ID Register* on page 4-84 |
| 1011 | 0xFCC | TRCDEVTYPE | RO | 0x00000013 | *4.29 TRCDEVTYPE, Device Type Register* on page 4-85 |
| 1012-1019 | 0xFD0-0xFEC | TRCPIDR0-7 | RO | - | *4.30 TRCPIDR0-7, Peripheral Identification Registers* on page 4-86 |
| 1020-1023 | 0xFF0-0xFFC | TRCCIDR0-3 | RO | - | *4.31 TRCCIDR0-3, Component Identification Registers* on page 4-88 |

——————— Note ———————

In *Table 4-1  ETM-M55 register summary* on page 4-37:

- The Reset value column shows the value of the register immediately after an ETM-M55 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.

## 4.2 TRCPRGCTLR, Programming Control Register

The TRCPRGCTLR enables ETM-M55.

**Usage constraints**

See *3.1.1 Controlling ETM-M55 programming* on page 3-34.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.

The following figure shows the TRCPRGCTLR bit assignments.



**Figure 4-1 TRCPRGCTLR bit assignments**

The following table shows the TRCPRGCTLR bit assignments.

**Table 4-2 TRCPRGCTLR bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:1] | - | RES0. |
| [0] | EN | ETM-M55 enable bit:<br>**0**        ETM-M55 is disabled.<br>**1**        ETM-M55 is enabled. |

## 4.3 TRCSTATR, Status Register

The TRCSTATR indicates the ETM-M55 status.

**Usage constraints**

There are no usage constraints.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.

The following figure shows the TRCSTATR bit assignments.
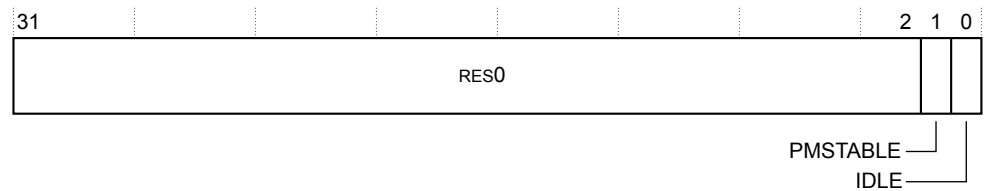


**Figure 4-2 TRCSTATR bit assignments**

The following table shows the TRCSTATR bit assignments.

**Table 4-3 TRCSTATR bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:2] | - | RES0. |
| [1] | PMSTABLE | Indicates whether the ETM-M55 registers are stable and can be read:<br><br>**0**      The programmers model is not stable.<br>**1**      The programmers model is stable. |
| [0] | IDLE | Indicates whether ETM-M55 is inactive:<br><br>**0**      ETM-M55 is not idle.<br>**1**      ETM-M55 is idle.<br><br>When the IDLE bit is set to 1:<br>• ETM-M55 is drained of any trace.<br>• Except for the programming interfaces, all external interfaces on ETM-M55 are quiescent. |

# 4.4 TRCCONFIGR, Trace Configuration Register

The TRCCONFIGR sets the basic tracing options for the ETM-M55.

**Usage constraints**

This register must always be programmed as part of ETM-M55 initialization.
Only accepts writes when ETM-M55 is disabled.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.

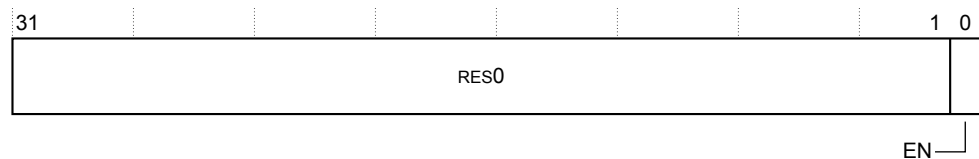The following figure shows the TRCCONFIGR bit assignments.
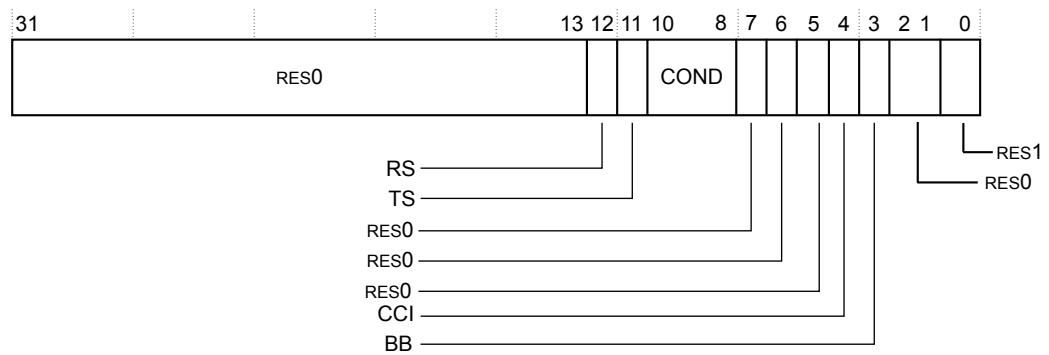


**Figure 4-3  TRCCONFIGR bit assignments**

The following table shows the TRCCONFIGR bit assignments.

**Table 4-4  TRCCONFIGR bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:13] | - | RES0. |
| [12] | RS | Return stack enable:<br><br>**0**     Return stack disabled.<br>**1**     Return stack enabled. |
| [11] | TS | Global timestamp tracing:<br><br>**0**     Global timestamp tracing disabled.<br>**1**     Global timestamp tracing enabled.<br><br>For more global timestamp tracing options, see *4.8 TRCTSCTLR, Global Timestamp Control Register* on page 4-47. |

**Table 4-4  TRCCONFIGR bit assignments (continued)**

| Bits | Name | Function |
|------|------|----------|
| [10:8] | COND | Conditional instruction tracing. The supported values are:<br><br>`0b000`     Conditional instruction tracing is disabled.<br>`0b001`     Conditional load instructions are traced.<br>`0b010`     Conditional store instructions are traced.<br>`0b011`     Conditional load and store instructions are traced.<br>`0b111`     All conditional instructions are traced.<br><br>All other values are Reserved. |
| [7] | VMID | RES0. |
| [6] | CID | RES0. |
| [5] | - | RES0. |
| [4] | CCI | Cycle counting in instruction trace:<br><br>**0**     Cycle counting in instruction trace disabled.<br>**1**     Cycle counting in instruction trace enabled.<br><br>For more cycle counting options, see *4.10 TRCCCCTLR, Cycle Count Control Register* on page 4-49. |
| [3] | BB | Branch broadcast mode:<br><br>**0**     Branch broadcast mode disabled.<br>**1**     Branch broadcast mode enabled. |
| [2:1] | INSTP0 | RES0 |
| [0] | - | RES1 |

## 4.5 TRCEVENTCTL0R, Event Control 0 Register

The TRCEVENTCTL0R controls the tracing of arbitrary events. The events also drive the ETM-M55 external outputs.

**Usage constraints**

This register must always be programmed as part of ETM-M55 initialization.
Only accepts writes when ETM-M55 is disabled.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37 .

The following figure shows the TRCEVENTCTL0R bit assignments.

| 31 | 16 | 15 | 8 | 7 | 0 |
|---|---|---|---|---|---|
| RES0 | | EVENT1 | | EVENT0 | |
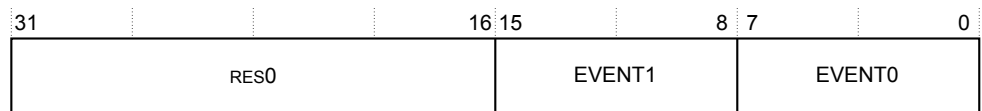
**Figure 4-4 TRCEVENTCTL0R bit assignments**

The following table shows the TRCEVENTCTL0R bit assignments.

**Table 4-5 TRCEVENTCTL0R bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:16] | - | RES0. |
| [15:8] | EVENT1 | Event selector 1. |
| [7:0] | EVENT0 | Event selector 0. |

## 4.6 TRCEVENTCTL1R, Event Control 1 Register

The TRCEVENTCTL1R controls how the events selected by TRCEVENTCTL0R behave.

See *4.5 TRCEVENTCTL0R, Event Control 0 Register* on page 4-44.

**Usage constraints**

This register must always be programmed as part of ETM-M55 initialization.
Only accepts writes when ETM-M55 is disabled.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.

The following figure shows the TRCEVENTCTL1R bit assignments.
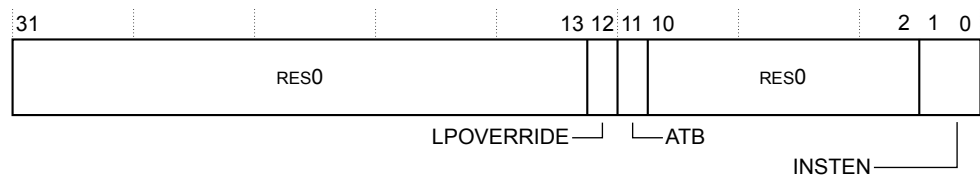


**Figure 4-5  TRCEVENTCTL1R bit assignments**

The following table shows the TRCEVENTCTL1R bit assignments.

**Table 4-6  TRCEVENTCTL1R bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:13] | - | RES0. |
| [12] | LPOVERRIDE | Low power state behavior override: <br><br> **0**      Low power state behavior unaffected. <br><br> **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: <br><br> **0**      ATB trigger disabled. <br><br> **1**      ATB trigger enabled. |
| [10:2] | - | RES0 |
| [1:0] | INSTEN | One bit per event, to enable generation of an event element in the instruction trace stream when the selected event occurs: <br><br> **0**      ETM-M55 does not generate an event element. <br><br> **1**      ETM-M55 generates an event element. |

## 4.7 TRCSTALLCTLR, Stall Control Register

The TRCSTALLCTLR enables ETM-M55 to stall the processor if the ETM-M55 FIFO goes over the programmed level to minimize risk of overflow.

**Usage constraints**

Only accepts writes when ETM-M55 is disabled.
This register must always be programmed as part of ETM-M55 initialization.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.

The following figure shows the TRCSTALLCTLR bit assignments.



**Figure 4-6 TRCSTALLCTLR bit assignments**

The following table shows the TRCSTALLCTLR bit assignments.

**Table 4-7 TRCSTALLCTLR bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:9] | - | RES0. |
| [8] | ISTALL | Stall processor based on instruction trace buffer space:<br><br>**0** ETM-M55 must not stall the processor.<br>**1** ETM-M55 can stall the processor.<br><br>The LEVEL field controls the threshold at which the processor is stalled. |
| [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:<br><br>`0b0000` Zero invasion. This setting has a greater risk of a FIFO overflow.<br>`0b0100` First level of invasion.<br>`0b1000` Second level of invasion.<br>`0b1100` Maximum invasion occurs, but there is less risk of a FIFO overflow.<br><br>——— **Note** ———<br>Writes to bits[1:0] are ignored and these bits are always set to `0b00`.<br><br>When the value of this field is `0b0100` or higher, then ETM-M55 might suppress the generation of:<br>• Periodic synchronization in the instruction trace stream.<br>• Global timestamps in the instruction trace stream.<br>• Cycle counting in the instruction trace stream, although the cumulative cycle count remains correct. |

## 4.8 TRCTSCTLR, 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 ETM-M55 is disabled.
This register must always be programmed as part of ETM-M55 initialization.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.
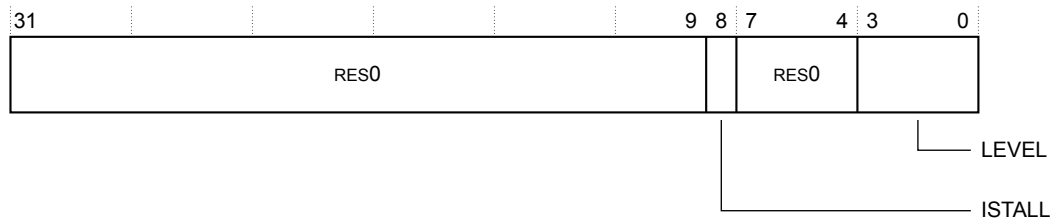
The following figure shows the TRCTSCTLR bit assignments.



| 31 | | | | | 8 | 7 | | 0 |
|----|--|--|--|--|---|---|--|---|
| RES0 | | | | | | EVENT | | |

**Figure 4-7 TRCTSCTLR bit assignments**

The following table shows the TRCTSCTLR bit assignments.

**Table 4-8 TRCTSCTLR bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:8] | - | RES0 |
| [7:0] | EVENT | An event selector. When the selected event is triggered, ETM-M55 inserts a global timestamp into the trace streams. For more information on the trace unit event that is selected, see the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4*. For more information on resource selection and event selectors, see *2.3.6 Resource selection* on page 2-30. |

## 4.9 TRCSYNCPR, Synchronization Period Register

TRCSYNCPR defines the number of bytes of trace between requests for trace synchronization. This specifies the period of trace synchronization of the trace streams.

**Usage constraints**

The register is implemented as RO and the synchronization period, which is indicated by the PERIOD bit field, is `0b01010`.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.

The following figure shows the TRCSYNCPR bit assignments.

| 31 | | | | | | 5 | 4 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | RES0 | | | | PERIOD | |
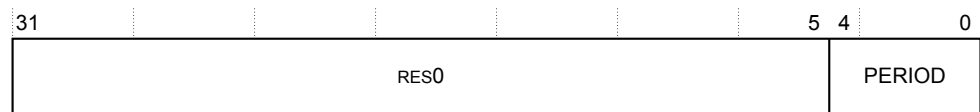
**Figure 4-8 TRCSYNCPR bit assignments**

The following table shows the TRCSYNCPR bit assignments.

**Table 4-9 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. For ETM-M55, this field is `0b01010`. |

## 4.10 TRCCCCTLR, 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 ETM-M55 is disabled.

This register must always be programmed as part of ETM-M55 initialization.

**Configurations**
Available in all configurations.

**Attributes**
See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.
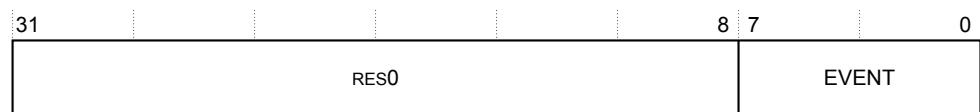
The following figure shows the TRCCCCTLR bit assignments.

| 31 | | | | 12 | 11 | | | 0 |
|---|---|---|---|---|---|---|---|---|
| | | RES0 | | | | | THRESHOLD | |

**Figure 4-9  TRCCCCTLR bit assignments**

The following table shows the TRCCCCTLR bit assignments.

**Table 4-10  TRCCCCTLR bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:12] | - | RES0 |
| [11:0] | THRESHOLD | Instruction trace cycle count threshold. If the value for this field is less than `0x4`, then the behavior is CONSTRAINED UNPREDICTABLE. |

## 4.11 TRCTRACEIDR, Trace ID Register

The TRCTRACEIDR sets the trace ID on the trace bus. Each trace source in a CoreSight system must be programmed with a unique non-reserved trace ID value so trace tools can distinguish trace from different sources.

**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 ETM-M55 initialization. Only accepts writes when ETM-M55 is disabled.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.

The following figure shows the TRCTRACEIDR bit assignments.
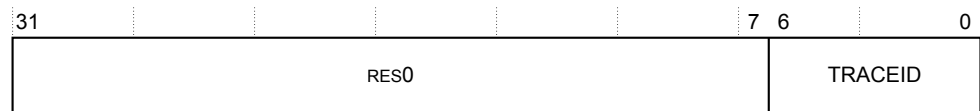


**Figure 4-10 TRCTRACEIDR bit assignments**

The following table shows the TRCTRACEIDR bit assignments.

**Table 4-11 TRCTRACEIDR bit assignments**

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

## 4.12 TRCVICTLR, ViewInst Main Control Register

The TRCVICTLR controls instruction trace filtering.

**Usage constraints**

Only accepts writes when ETM-M55 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 4-1 ETM-M55 register summary* on page 4-37.

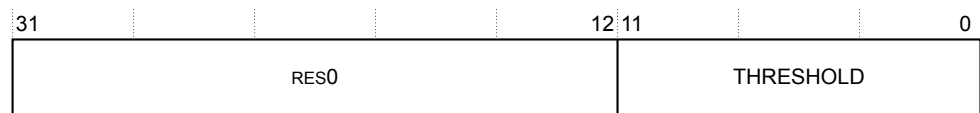The following figure shows the TRCVICTLR bit assignments.



**Figure 4-11 TRCVICTLR bit assignments**

The following table shows the TRCVICTLR bit assignments.

**Table 4-12 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: <br><br> **0** ETM-M55 does not generate instruction trace, in Secure state, for exception level n. <br><br> **1** ETM-M55 generates instruction trace, in Secure state, for exception level n. <br><br> The exception levels are: <br><br> **Bit[16]** Thread mode. <br> **Bit[17]** RES0. <br> **Bit[18]** RES0. EXLEVEL_S[2] is never implemented. <br> **Bit[19]** Handler mode. |
| [15:12] | - | RES0. |
| [11] | TRCERR | Selects whether a system error exception must always be traced: <br><br> **0** System error exception is traced only if the instruction or exception immediately before the system error exception is traced. <br><br> **1** System error exception is always traced, regardless of the value of ViewInst. |

**Table 4-12  TRCVICTLR bit assignments (continued)**

| Bits | Name | Function |
|------|------|----------|
| [10] | TRCRESET | Selects whether a reset exception must always be traced:<br><br>**0**       Reset exception is traced only if the instruction or exception immediately before the reset exception is traced.<br><br>**1**       Reset exception is always traced regardless of the value of ViewInst. |
| [9] | SSSTATUS | Indicates the current status of the start/stop logic:<br><br>**0**       Start/stop logic is in the stopped state.<br><br>**1**       Start/stop logic is in the started state. |
| [8] | - | RES0 |
| [7:0] | EVENT | An event selector.<br><br>For more information on the trace unit event that is selected, see the *Arm® Embedded Trace Macrocell Architecture Specification ETMv4*. For more information on resource selection and event selectors, see *2.3.6 Resource selection* on page 2-30. |

## 4.13  TRCVIPCSSCTLR, ViewInst Start/Stop Processor Comparator Control Register

The TRCVIPCSSCTLR is used to set or read which processor comparator inputs can control the ViewInst start/stop logic.

**Usage constraints**

Only accepts writes when ETM-M55 is disabled.
This register must be programmed.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.

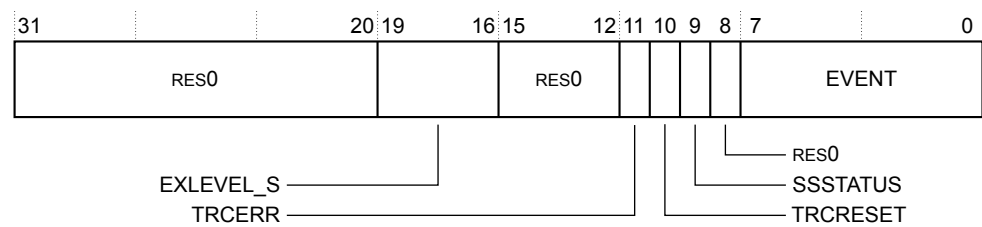The following figure shows the TRCVIPCSSCTLR bit assignments when there are two processor comparator inputs implemented for the *Data Watchpoint Trigger* (DWT). TRCIDR4.NUMPC is `0b0010`. For more information on TRCIDR4, see *4.16.5 TRCIDR4, ID Register 4* on page 4-61.



**Figure 4-12  TRCVIPCSSCTLR bit assignments for two processor comparator inputs**

The following table shows the TRCVIPCSSCTLR bit assignments for two processor comparator inputs.

**Table 4-13  TRCVIPCSSCTLR bit assignments for two processor comparator inputs**

| Bits | Name | Function |
|---|---|---|
| [31:18] | - | RES0 |
| [17:16] | STOP | Selects which processor comparator inputs are used with ViewInst start/stop control, for the purpose of stopping trace. Each bit represents a processor comparator input, so bit[m] controls the selection of processor comparator input m-16. If bit[m] is: <br><br> **0**      The single processor comparator input m-16 is not selected as a stop resource. <br><br> **1**      The single processor comparator input m-16 is selected as a stop resource. |
| [15:2] | - | RES0. |
| [1:0] | START | Selects which processor comparator inputs are used with ViewInst start/stop control, for the purpose of starting trace. Each bit represents a processor comparator input, so bit[n] controls the selection of processor comparator. If bit[n] is: <br><br> **0**      The single processor comparator input n is not selected as a start resource. <br><br> **1**      The single processor comparator input n is selected as a start resource. |

The following figure shows the TRCVICTLR bit assignments when there are four processor comparator inputs implemented for the DWT. TRCIDR4.NUMPC is `0b0100`. For more information on TRCIDR4, see *4.16.5 TRCIDR4, ID Register 4* on page 4-61.

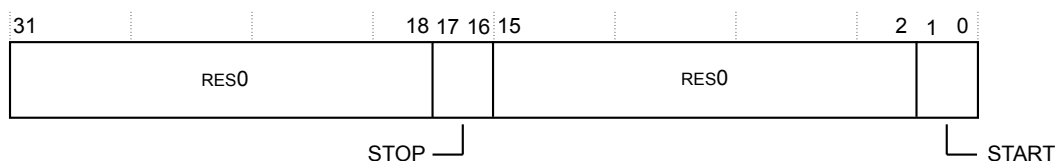| 31 | 20 | 19 | 16 | 15 | 4 | 3 | 0 |
|----|----|----|----|----|---|---|---|
| RES0 | | STOP | | RES0 | | START | |

**Figure 4-13  TRCVIPCSSCTLR bit assignments for four processor comparator inputs**

The following table shows the TRCVIPCSSCTLR bit assignments for four processor comparator inputs.

**Table 4-14  TRCVIPCSSCTLR bit assignments for four processor comparator inputs**

| Bits | Name | Function |
|------|------|----------|
| [31:20] | - | RES0 |
| [19:16] | STOP | Selects which processor comparator inputs are used with ViewInst start/stop control, for the purpose of stopping trace. Each bit represents a processor comparator input, so bit[m] controls the selection of processor comparator input m-16. If bit[m] is:<br><br>**0**  The single processor comparator input m-16 is not selected as a stop resource.<br>**1**  The single processor comparator input m-16 is selected as a stop resource. |
| [15:4] | - | RES0. |
| [3:0] | START | Selects which processor comparator inputs are used with ViewInst start/stop control, for the purpose of starting trace. Each bit represents a processor comparator input, so bit[n] controls the selection of processor comparator. If bit[n] is:<br><br>**0**  The single processor comparator input n is not selected as a start resource.<br>**1**  The single processor comparator input n is selected as a start resource. |

## 4.14 TRCEXTINSELR, External Input Select Register

The TRCEXTINSELR is used to set or read which external inputs are resources to ETM-M55.

**Usage constraints**

Only accepts writes when ETM-M55 is disabled.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.

The following figure shows the TRCEXTINSELR bit assignments.

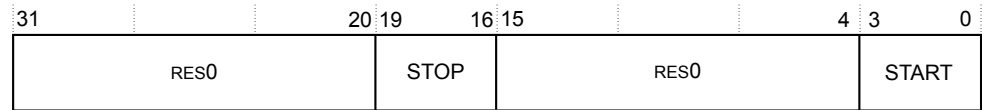| 31 | 24 23 | 16 15 | 8 7 | 0 |
|---|---|---|---|---|
| SEL3 | SEL2 | SEL1 | SEL0 |
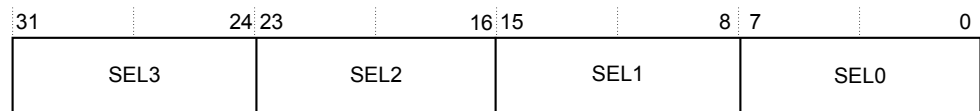
**Figure 4-14  TRCEXTINSELR bit assignments for two processor comparator inputs**

The following table shows the TRCEXTINSELR bit assignments.

**Table 4-15  TRCEXTINSELR bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:24] | SEL3 | This field is a binary value, of up to 8 bits, that selects which external input is a resource for ETM-M55. |
| [23:16] | SEL2 | This field is a binary value, of up to 8 bits, that selects which external input is a resource for ETM-M55. |
| [15:8] | SEL1 | This field is a binary value, of up to 8 bits, that selects which external input is a resource for ETM-M55. |
| [7:0] | SEL0 | This field is a binary value, of up to 8 bits, that selects which external input is a resource for ETM-M55. |

## 4.15    TRCCNTRLDVR0, Counter Reload Value Register 0

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

**Usage constraints**

Only accepts writes when ETM-M55 is disabled.
The count value is only stable when TRCSTATR.PMSTABLE is 1.
If software uses counter 0, 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 4-1  ETM-M55 register summary* on page 4-37.

The following figure shows the TRCCNTRLDVR0 bit assignments.



| 31 | | | 16 | 15 | | | 0 |
|---|---|---|---|---|---|---|---|
| | RES0 | | | | VALUE | | |

**Figure 4-15  TRCCNTRLDVR0 bit assignments**

The following table shows the TRCCNTRLDVR0 bit assignments.

**Table 4-16  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. |

## 4.16 TRCIDR0-13, ID Registers

This section describes the ETM-M55 ID registers.

### 4.16.1 TRCIDR0, ID Register 0

The TRCIDR0 indicates the tracing capabilities of the ETM-M55 instruction trace.

**Usage constraints**

This register is read-only.

**Configurations**

This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.

The following figure shows the TRCIDR0 bit assignments.



**Figure 4-16  TRCIDR0 bit assignments**

The following table shows the TRCIDR0 bit assignments.

**Table 4-17  TRCIDR0 bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:30] | - | RES0. |
| [29] | COMMOPT | Indicates the meaning of the commit field in some packets:<br><br>**1**       Commit mode 1. |
| [28:24] | TSSIZE | Global timestamp size:<br><br>`0b01000`      Maximum of 64-bit global timestamp implemented. |
| [23:20] | - | Reserved, RES0. |
| [19:18] | BF | Branch Future Support.<br><br>`0b00`      Branch future not supported. |
| [17] | TRCEXDATA | Indicates support for the tracing of data transfers for exceptions and exception returns:<br><br>**0**      TRCVDCTLR.TRCEXDATA is not implemented. |

**Table 4-17  TRCIDR0 bit assignments (continued)**

| Bits | Name | Function |
|------|------|----------|
| [16:15] | QSUPP | Indicates Q element support:<br><br>0b00          Q elements not supported. |
| [14] | QFILT | RES0. |
| [13:12] | CONDTYPE | Indicates how conditional results are traced:<br><br>0b00    ETM-M55 indicates only if a conditional instruction passes or fails its condition code check. |
| [11:10] | NUMEVENT | Number of events supported in the trace:<br><br>0b01     Two events supported. |
| [9] | RETSTACK | Return stack support:<br><br>1      Two entry return stack implemented. |
| [8] | - | RES0. |
| [7] | TRCCCI | Support for cycle counting in the instruction trace:<br><br>1    Cycle counting in the instruction trace is implemented. |
| [6] | TRCCOND | Support for conditional instruction tracing:<br><br>1     Conditional instruction tracing is implemented. |
| [5] | TRCBB | Support for branch broadcast tracing:<br><br>1     Branch broadcast tracing is implemented. |
| [4:3] | TRCDATA | Support for tracing of data:<br><br>0b00          Data tracing is not supported. |
| [2:1] | INSTP0 | Support for tracing of load and store instructions as P0 elements:<br><br>0b00     Tracing of load and store instructions as P0 elements is not supported. |
| [0] | - | RES1 |

## 4.16.2 TRCIDR1, ID Register 1

The TRCIDR1 indicates the ETM-M55 architecture.

**Usage constraints**

This register is read-only.

**Configurations**

This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.
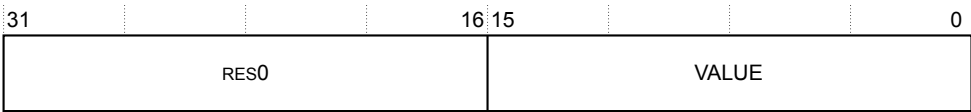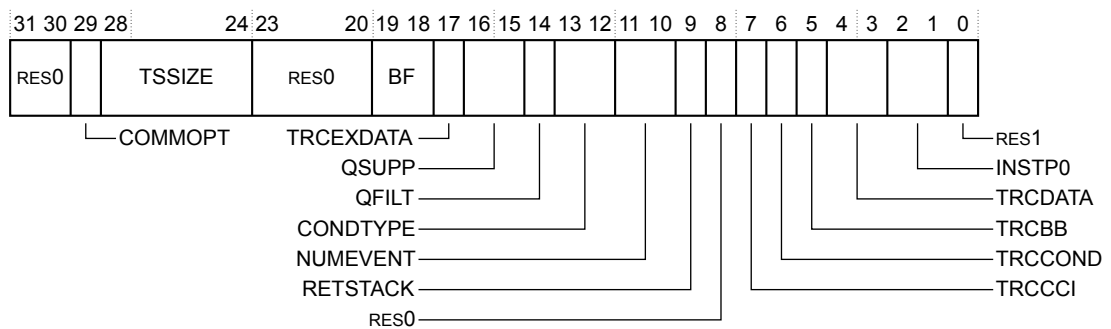
The following figure shows the TRCIDR1 bit assignments.

**Figure 4-17  TRCIDR1 bit assignments**

The following table shows the TRCIDR1 bit assignments.

**Table 4-18  TRCIDR1 bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:24] | DESIGNER | Indicates the designer of the trace unit:<br><br>`0x41`                                                      Arm |
| [23:16] | - | RES0. |
| [15:12] | - | RES1. |
| [11:8] | TRCARCHMAJ | Major ETM-M55 architecture version number:<br><br>`0x4`                                      ETMv4. |
| [7:4] | TRCARCHMIN | Minor ETM-M55 architecture version number:<br><br>`0x5`                          Minor revision 5. |
| [3:0] | REVISION | Implementation revision number:<br><br>`0x0`            Implementation revision `0x0`. |

### 4.16.3    TRCIDR2, ID Register 2

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

**Usage constraints**
>        This register is read-only.

**Configurations**
>        This register is available in all configurations.

**Attributes**
>        See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.
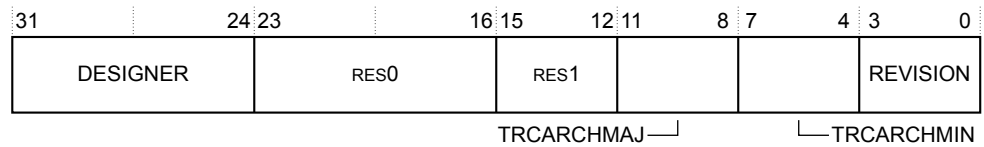
The following figure shows the TRCIDR2 bit assignments.



**Figure 4-18  TRCIDR2 bit assignments**

The following table shows the TRCIDR2 bit assignments.

**Table 4-19  TRCIDR2 bit assignments**

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

### 4.16.4    TRCIDR3, ID Register 3

The TRCIDR3 indicates certain aspects of the ETM-M55 configuration.

**Usage constraints**

This register is read-only.

**Configurations**

This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.
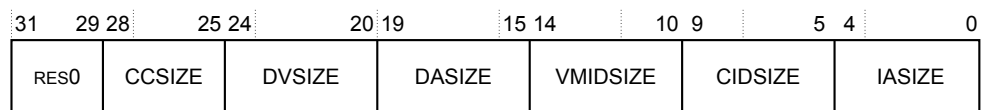
The following figure shows the TRCIDR3 bit assignments.



**Figure 4-19  TRCIDR3 bit assignments**

The following table shows the TRCIDR3 bit assignments.

**Table 4-20  TRCIDR3 bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31] | NOOVERFLOW | Indicates whether TRCSTALLCTLR.NOOVERFLOW is implemented:<br><br>**0**     NOOVERFLOW is not implemented. |
| [30:28] | NUMPROC | Indicates the number of processors available for tracing.<br><br>`0b000`               ETM-M55 can trace one processor.<br><br>NUMPROC uses bits [30:28] and bits [13:12] to form a single 5-bit field.Bits [13:12] form the top bits of this field. |
| [27] | SYSSTALL | System support for stall control of the processor.<br><br>**1**               System supports stall control of the processor.<br><br>This field is used with STALLCTL. The system supports stalling of the processor only when both SYSSTALL and STALLCTL are `1` . |
| [26] | STALLCTL | Stall control support:<br><br>**1**      TRCSTALLCTLR is implemented. |
| [25] | SYNCPR | Indicates trace synchronization period support:<br><br>**1**   TRCSYNCPR is read-only for instruction trace only configuration. The trace synchronization period is fixed. |
| [24] | TRCERR | Indicates whether TRCVICTLR.TRCERR is implemented:<br><br>**1**               TRCERR is implemented. |
| [23:20] | EXLEVEL_NS | RES0 |
| [19:16] | EXLEVEL_S | Exception levels implemented. One bit for each level.<br><br>`0b1001`             Privilege levels Thread and Handler are implemented.<br><br>Thread is at exception level 0 and Handler is at exception level 3. |
| [15:14] | - | RES0. |
| [13:12] | NUMPROC | Indicates the number of processors available for tracing.<br><br>`0b00`               ETM-M55 can trace one processor.<br><br>NUMPROC uses bits [30:28] and bits [13:12] to form a single 5-bit field.Bits [13:12] form the top bits of this field. |
| [11:0] | CCITMIN | Minimum value which can be programmed to TRCCCCTLR.THRESHOLD, defining the minimum cycle counting threshold.<br><br>`0x4`      Minimum of four instruction trace cycles. |

### 4.16.5    TRCIDR4, ID Register 4

The TRCIDR4 indicates the available ETM-M55 resources.

**Usage constraints**
This register is read-only

**Configurations**
This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.
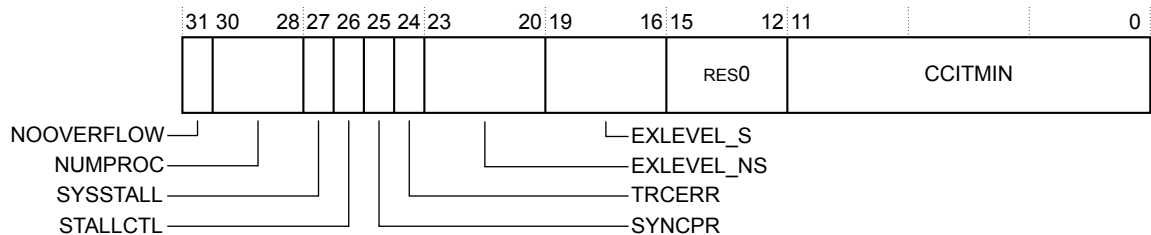
The following figure shows the TRCIDR4 bit assignments.



**Figure 4-20  TRCIDR4 bit assignments**

The following table shows the TRCIDR4 bit assignments.

**Table 4-21  TRCIDR4 bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:28] | NUMVMIDC | Number of *Virtual Machine ID* (VMID) comparators implemented:<br><br>0b0000      VMID comparators are not implemented. |
| [27:24] | NUMCIDC | Number of Context ID comparators implemented:<br><br>0b0000      Context ID comparators are not supported. |
| [23:20] | NUMSSCC | Number of single-shot comparator controls implemented:<br><br>0b0001      One single-shot comparator control is implemented. |
| [19:16] | NUMRSPAIR | Number of resource selection pairs implemented:<br><br>0b0001      Two resource selection pairs are implemented. |
| [15:12] | NUMPC | Number of processor comparator inputs implemented for the DWT:<br><br>0b0010      Two processor comparator inputs.<br>0b0100      Four processor comparator inputs. |
| [11:9] | - | RES0 |
| [8] | SUPPDAC | Data address comparisons implemented:<br><br>0      Data address comparisons are not supported. |
| [7:4] | NUMDVC | Number of data value comparators implemented:<br><br>0b0000      No data value comparators are implemented. |
| [3:0] | NUMACPAIRS | Number of address comparator pairs implemented:<br><br>0b0000      No address comparator pairs are implemented. |

### 4.16.6    TRCIDR5, ID Register 5

The TRCIDR5 indicates the available ETM-M55 resources.

**Usage constraints**

This register is read-only.

**Configurations**

This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.
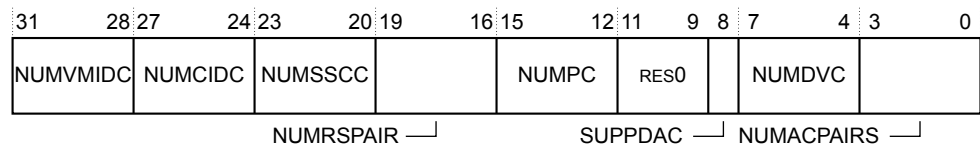
The following figure shows the TRCIDR5 bit assignments.



**Figure 4-21  TRCIDR5 bit assignments**

The following table shows the TRCIDR5 bit assignments.

**Table 4-22  TRCIDR5 bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31] | REDFUNCNTR | Reduced Function Counter implemented:<br><br>**1**       Counter 0 is implemented as a Reduced Function Counter. |
| [30:28] | NUMCNTR | Number of counters implemented:<br><br>`0b001`    One counter implemented. |
| [27:25] | NUMSEQSTATE | Number of sequencer states implemented:<br><br>`0b000`    No sequencer states implemented. |
| [24] | - | RES0 |
| [23] | LPOVERRIDE | Low-power state override support:<br><br>**1**       Low-power state override support implemented. |
| [22] | ATBTRIG | ATB trigger support:<br><br>**1**       ATB trigger support implemented. |
| [21:16] | TRACEIDSIZE | Number of bits of trace ID:<br><br>`0x07`    7-bit trace ID implemented. |
| [15:12] | - | RES0. |
| [11:9] | NUMEXTINSEL | Number of input selectors implemented.<br><br>`0b100`    Four input selectors, SEL0 to SEL3, are present. |
| [8:0] | NUMEXTIN | Number of external inputs implemented:<br><br>`0x4+` number of *Performance Monitor Unit* (PMU) events      Four external inputs and any PMU event inputs are implemented. |

### 4.16.7    TRCIDR6, ID Register 6

The TRCIDR6 is reserved, RES0.

### 4.16.8    TRCIDR7, ID Register 7

The TRCIDR7 is reserved, RES0.

#### 4.16.9 TRCIDR8, ID Register 8

The TRCIDR8 indicates the maximum speculation depth of the instruction trace stream.

**Usage constraints**

This register is read-only.

**Configurations**

This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.
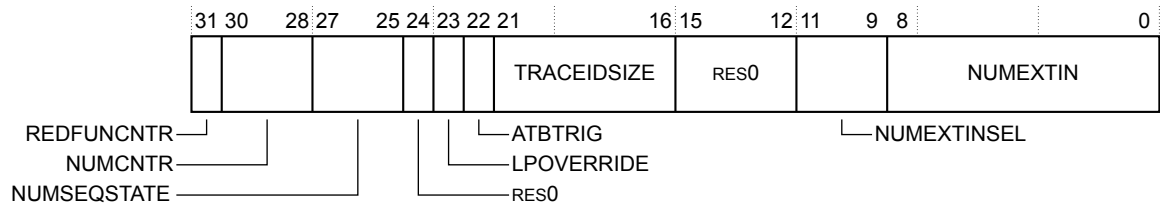
The following figure shows the TRCIDR8 bit assignments.

```
31                                                        0
┌──────────────────────────────────────────────────────┐
│                      MAXSPEC                           │
└──────────────────────────────────────────────────────┘
```

**Figure 4-22 TRCIDR8 bit assignments**

The following table shows the TRCIDR8 bit assignments.

**Table 4-23 TRCIDR8 bit assignments**

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

#### 4.16.10 TRCIDR9, ID Register 9

The TRCIDRn indicates the number of P0 right-hand keys that are used.

**Usage constraints**

This register is read-only.

**Configurations**

This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.
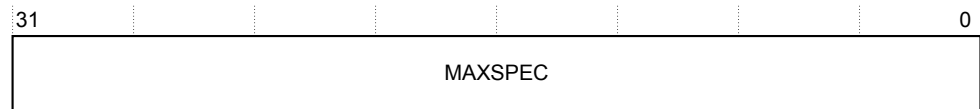
The following figure shows the TRCIDR9 bit assignments.

```
31                                                        0
┌──────────────────────────────────────────────────────┐
│                      NUMP0KEY                          │
└──────────────────────────────────────────────────────┘
```

**Figure 4-23 TRCIDR9 bit assignments**

The following table shows the TRCIDR9 bit assignments.

**Table 4-24 TRCIDR9 bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:0] | NUMP0KEY | `0x00000000`      No P0 keys used in instruction trace. |

#### 4.16.11 TRCIDR10, ID Register 10

The TRCIDR10 indicates the total number of P1 right-hand keys, including normal and special keys.

**Usage constraints**

This register is read-only.

**Configurations**

This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.
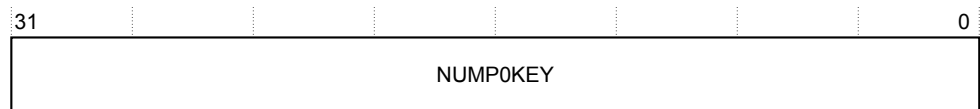
The following figure shows the TRCIDR10 bit assignments.

| 31 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|
| | | | NUMP1KEY | | | | |

**Figure 4-24 TRCIDR10 bit assignments**

The following table shows the TRCIDR10 bit assignments.

**Table 4-25 TRCIDR10 bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:0] | NUMP1KEY | `0x00000000`        No P1 right-hand keys used in instruction trace. |

#### 4.16.12 TRCIDR11, ID Register 11

The TRCIDR11 indicates the number of special P1 right-hand keys.

**Usage constraints**

This register is read-only

**Configurations**

This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.
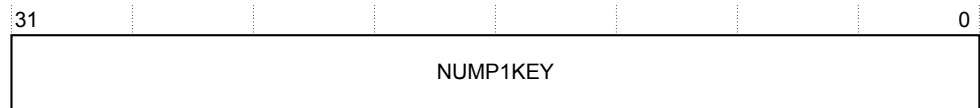
The following figure shows the TRCIDR11 bit assignments.

| 31 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|
| | | | NUMP1SPC | | | | |

**Figure 4-25 TRCIDR11 bit assignments**

The following table shows the TRCIDR11 bit assignments.

**Table 4-26 TRCIDR11 bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:0] | NUMP1SPC | `0x00000000`        No special P1 right-hand keys used in any configuration. |

#### 4.16.13 TRCIDR12, ID Register 12

The TRCIDR12 indicates the total number of conditional instruction right-hand keys, including normal and special keys.

**Usage constraints**

This register is read-only.

**Configurations**

This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.
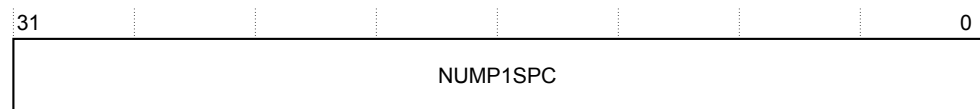
The following figure shows the TRCIDR12 bit assignments.



**Figure 4-26  TRCIDR12 bit assignments**

The following table shows the TRCIDR12 bit assignments.

**Table 4-27  TRCIDR12 bit assignments**

| Bits | Name | Function | |
| --- | --- | --- | --- |
| [31:0] | NUMCONDKEY | 0x00000001 | One conditional instruction right-hand key implemented. |

### 4.16.14   TRCIDR13, ID Register 13

The TRCIDR13 indicates the number of special conditional instruction right-hand keys.

**Usage constraints**

This register is read-only.

**Configurations**

This register is available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.
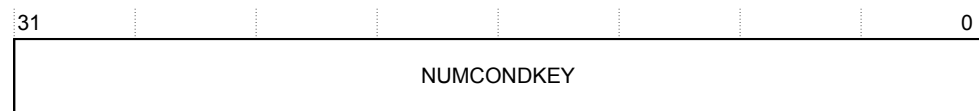
The following figure shows the TRCIDR13 bit assignments.



**Figure 4-27  TRCIDR13 bit assignments**

The following table shows the TRCIDR13 bit assignments.

**Table 4-28  TRCIDR13 bit assignments**

| Bits | Name | Function | |
| --- | --- | --- | --- |
| [31:0] | NUMCONDSPC | 0x00000000 | No special conditional instruction right-hand keys implemented. |

## 4.17　TRCRSCTLRn, Resource Selection Registers 2-3

The TRCRSCTLRn controls the selection of trace resources.

**Usage constraints**

Only accepts writes when ETM-M55 is disabled.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.

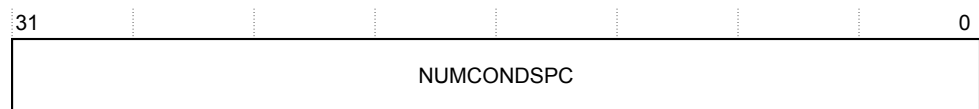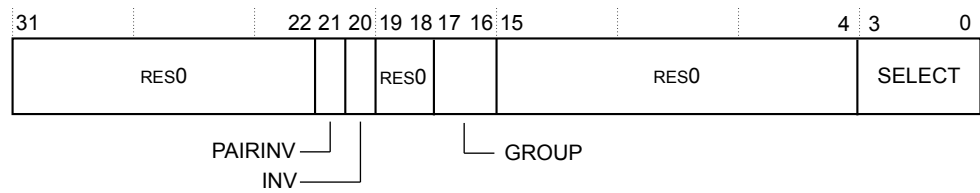The following figure shows the TRCRSCTLRn bit assignments.



**Figure 4-28　TRCRSCTLRn bit assignments**

The following table shows the TRCRSCTLRn bit assignments.

**Table 4-29　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 TRCRSCTLR2 and not on TRCRSCTLR3. |
| [20] | INV | Inverts the selected resources:<br>**0**　　Resource is not inverted.<br>**1**　　Resource is inverted. |
| [19:18] | - | RES0. |
| [17:16] | GROUP | Selects a group of resources. |
| [15:4] | - | RES0. |
| [3:0] | SELECT | Selects one or more resources from the wanted group. One bit is provided per resource from the group. |

The following table lists which resources are selected, depending on the values of the GROUP and SELECT fields.

**Table 4-30　Resource selection**

| Group | Select | Resource |
|---|---|---|
| 0b0000 | Bits 0-3 | External input selectors 0-3. When select bit N is set, then the resource selector is sensitive to external input selector N. |
| 0b0001 | Bits 0-3 | When select bit N is set, the resource selector is sensitive to processor comparator input N. |

**Table 4-30  Resource selection (continued)**

| Group | Select | Resource |
|---|---|---|
| 0b0010 | 0 | Counter at zero 0 |
| 0b0011 | 0 | Single-Shot Comparator control 0 |
| 0b0100-0b1111 | 0-15 | Reserved |

## 4.18 TRCSSCCR0, Single-shot Comparator Control Register 0

The TRCSSCCR0 controls the single-shot comparator.

**Usage constraints**

Only accepts writes when ETM-M55 is disabled.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.
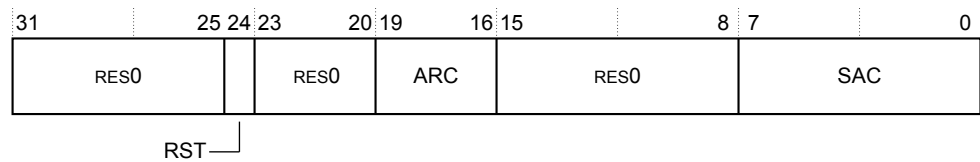
The following figure shows the TRCSSCCR0 bit assignments.



**Figure 4-29 TRCSSCCR0 bit assignments**

The following table shows the TRCSSCCR0 bit assignments.

**Table 4-31 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: <br><br> **0** When the single-shot comparator resource fires, it is not reset. <br><br> **1** When the single-shot comparator resource fires, it is reset. This enables the single-shot comparator resource to fire multiple times. |
| [23:20] | - | RES0. |
| [19:16] | ARC | RAZ/WI. |
| [15:8] | - | RES0. |
| [7:0] | SAC | RAZ/WI. |

## 4.19 TRCSSCSR0, Single-shot Comparator Status Register 0

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

**Usage constraints**

Only accepts writes when ETM-M55 is disabled.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.
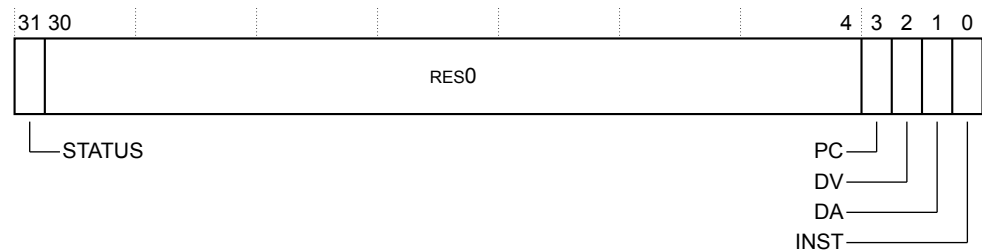
The following figure shows the TRCSSCSR0 bit assignments.



**Figure 4-30 TRCSSCSR0 bit assignments**

The following table shows the TRCSSCSR0 bit assignments.

**Table 4-32 TRCSSCSR0 bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31] | STATUS | Single-shot status. This indicates whether any of the selected comparators have matched:<br><br>**0**       Match has not occurred.<br>**1**       Match has occurred at least once.<br><br>When programming ETM-M55, if TRCSSCCR0.RST is 0, the STATUS bit must be explicitly written to 0 to enable this single-shot comparator control.<br><br>If TRCSSCCR0.RST is set, the STATUS bit clears automatically so it is not necessary to clear the STATUS bit when programming. Otherwise, the STATUS bit must be cleared when programming ETM-M55 otherwise a single-shot comparison cannot occur. |
| [30:4] | - | RES0 |
| [3] | PC | Indicates that the Single-shot comparator is sensitive to processor comparator inputs:<br><br>**1**       Single-shot comparator is sensitive to processor comparator inputs. |
| [2] | DV | Data value comparator support:<br><br>**0**       Single-shot data value comparisons not supported. |

**Table 4-32  TRCSSCSR0 bit assignments (continued)**

| Bits | Name | Function |
|------|------|----------|
| [1] | DA | Data address comparator support:<br><br>**0**      Single-shot data address comparisons not supported. |
| [0] | INST | Instruction address comparator support:<br><br>**0**      Single-shot instruction address comparisons not supported. |

## 4.20 TRCSSPCICR0, Single-shot Processor Comparator Input Control Register 0

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

**Usage constraints**

Only accepts writes when ETM-M55 is disabled.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.

The following figure shows the TRCSSPCICR0 bit assignments.

| 31 | | | | | | 4 | 3 | 0 |
|---|---|---|---|---|---|---|---|---|
| RES0 | | | | | | | PC | |

**Figure 4-31 TRCSSPCICR0 bit assignments**

The following table shows the TRCSSPCICR0 bit assignments.

**Table 4-33 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. |

## 4.21 TRCPDCR, Power Down Control Register

The TRCPDCR request to the system power controller to keep ETM-M55 powered up.

**Usage constraints**

There are no usage constraints.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.
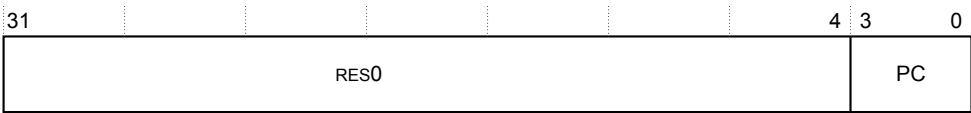
The following figure shows the TRCPDCR bit assignments.



**Figure 4-32  TRCPDCR bit assignments**

The following table shows the TRCPDCR bit assignments.

**Table 4-34  TRCPDCR bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:4] | - | RES0 |
| [3] | PU | Power up request, to request that power to ETM-M55 and access to the trace registers is maintained:<br>**0**    Power not requested.<br>**1**    Power requested.<br><br>This bit is reset to 0 on ETM-M55 reset. |
| [2:0] | - | RES0 |

## 4.22 TRCPDSR, Power Down Status Register

The TRCPDSR indicates the power down status of the ETM-M55.

**Usage constraints**

There are no usage constraints.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.
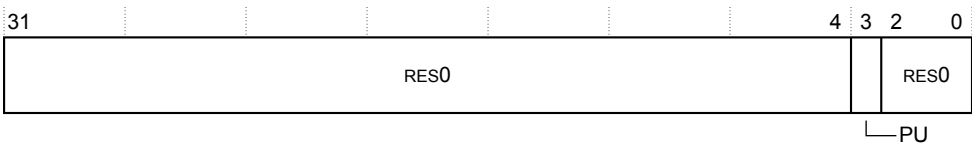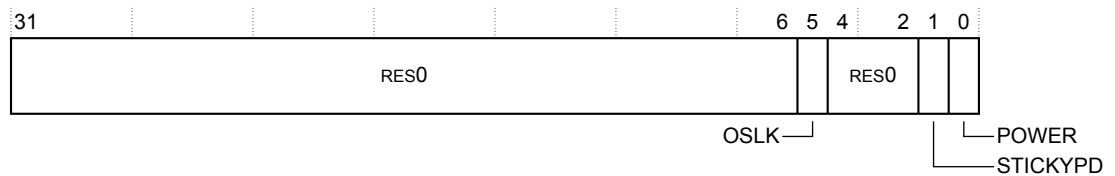
The following figure shows the TRCPDSR bit assignments.



**Figure 4-33  TRCPDSR bit assignments**

The following table shows the TRCPDSR bit assignments.

**Table 4-35  TRCPDSR bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:6] | - | RES0 |
| [5] | OSLK | RES0 |
| [4:2] | - | RES0 |
| [1] | STICKYPD | Sticky power down state.<br><br>**0**       Trace register power has not been removed since the TRCPDSR was last read.<br><br>**1**       Trace register power has been removed since the TRCPDSR was last read.<br><br>This bit is set to 1 when power to the ETM-M55 registers is removed, to indicate that programming state has been lost. It is cleared after a read of the TRCPDSR. |
| [0] | POWER | Indicates ETM-M55 is powered up:<br><br>**1**       ETM-M55 is powered up. All registers are accessible.<br><br>If a system implementation allows ETM-M55 to be powered down independently of the debug power domain, the system must ensure:<br>• Accesses to ETM-M55 complete correctly.<br>• Reads to this location return 0 to indicate that ETM-M55 can be powered down. |

## 4.23 Integration test registers

The ETM-M55 integration test registers 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 *4.23.5 TRCITCTRL, Integration Mode Control Register* on page 4-79 to 1.

- You can use the write-only integration test registers to set the outputs of some of the ETM-M55 signals. The following table shows the signals that can be controlled in this way.

**Table 4-36  Output signals that the integration test registers can control**

| Signal | Register | Bits | Register description |
|---|---|---|---|
| **AFREADYE** | TRCITIATBOUTR | [1] | *4.23.4 TRCITIATBOUTR, Integration Instruction ATB Out Register* on page 4-78 |
| **ATIDE[6:0]** | TRCITATBIDR | [6:0] | *4.23.1 TRCITATBIDR, Integration ATB Identification Register* on page 4-76 |
| **ATDATAE[6:0]** | TRCITIDATAR | [6:0] | *4.23.2 TRCITIDATAR, Integration Data Register* on page 4-77 |
| **ATVALIDE** | TRCITIATBOUTR | [0] | *4.23.4 TRCITIATBOUTR, Integration Instruction ATB Out Register* on page 4-78 |

- You can use the read-only integration test registers to read the state of some of the ETM-M55 input signals. The following table shows the signals that can be read in this way.

**Table 4-37  Input signals that the integration test registers can read**

| Signal | Register | Bits | Register description |
|---|---|---|---|
| **AFVALIDE** | TRCITIATBINR | [1] | *4.23.3 TRCITIATBINR, Integration Instruction ATB In Register* on page 4-77 |
| **ATREADYE** | TRCITIATBINR | [0] | *4.23.3 TRCITIATBINR, Integration Instruction ATB In Register* on page 4-77 |

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

### 4.23.1    TRCITATBIDR, Integration ATB Identification Register

The TRCITATBIDR sets the state of output pins.

The output pins are listed in *Table 4-38  TRCITATBIDR bit assignments* on page 4-76.

**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.
- This is a write-only register.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37 and *Table 4-36  Output signals that the integration test registers can control* on page 4-75.

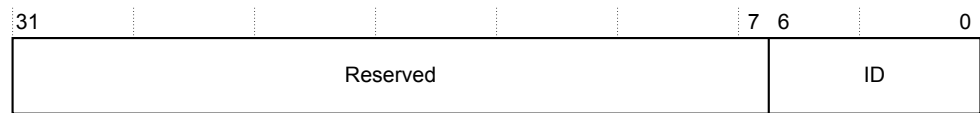The following figure shows the TRCITATBIDR bit assignments.



**Figure 4-34  TRCITATBIDR bit assignments**

The following table shows the TRCITATBIDR bit assignments.

**Table 4-38  TRCITATBIDR bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:7] | - | RES0. |
| [6:0] | ID | Drives the **ATIDE[6:0]** output pin. |

### 4.23.2 TRCITIDATAR, Integration Data Register

The TRCITIDATAR sets the state of output pins.

The output pins are listed in *Table 4-39 TRCITIDATAR bit assignments* on page 4-77.

**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.
- This is a write-only register.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37 and *Table 4-36 Output signals that the integration test registers can control* on page 4-75.

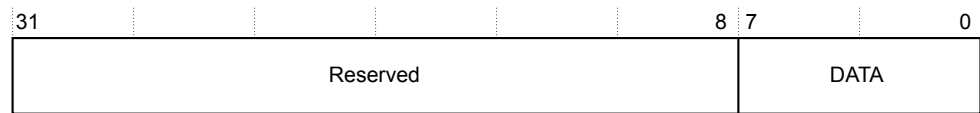The following figure shows the TRCITIDATAR bit assignments.

**Figure 4-35 TRCITIDATAR bit assignments**

The following table shows the TRCITIDATAR bit assignments.

**Table 4-39 TRCITIDATAR bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:8] | - | RES0. |
| [7:0] | DATA | Drives the **ATDATAE[7:0]** output pin. |

### 4.23.3 TRCITIATBINR, Integration Instruction ATB In Register

The TRCITIATBINR reads the state of the input pins.

The input pins are listed in *Table 4-40 TRCITIATBINR bit assignments* on page 4-78.

**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 4-1 ETM-M55 register summary* on page 4-37 and *Table 4-36 Output signals that the integration test registers can control* on page 4-75.

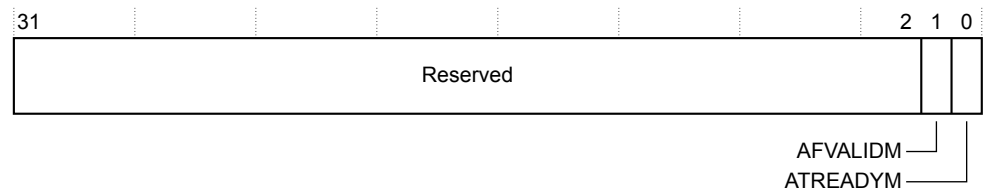The following figure shows the TRCITIATBINR bit assignments.

**Figure 4-36  TRCITIATBINR bit assignments**

The following table shows the TRCITIATBINR bit assignments.

**Table 4-40  TRCITIATBINR bit assignments**

| Bits | Name | Function |
|---|---|---|
| [31:2] | - | Reserved. Read undefined. |
| [1] | AFVALIDM | Returns the value of the **AFVALIDE** input pin[b]. |
| [0] | ATREADYM | Returns the value of the **ATREADYE** input pin. |

### 4.23.4  TRCITIATBOUTR, Integration Instruction ATB Out Register

The TRCITIATBOUTR sets the state of the output pins.

These output pins are listed in*Table 4-41  TRCITIATBOUTR bit assignments* on page 4-79.

**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.
- This is a write-only register.

**Configurations**
Available in all configurations.

**Attributes**
See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37 and *Table 4-36  Output signals that the integration test registers can control* on page 4-75.

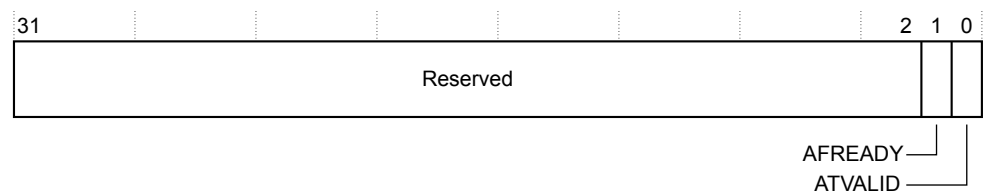The following figure shows the TRCITIATBOUTR bit assignments.



**Figure 4-37  TRCITIATBOUTR bit assignments**

The following table shows the TRCITIATBOUTR bit assignments.

---

[b]   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.

**Table 4-41 TRCITIATBOUTR bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:2] | - | Reserved. Read undefined. |
| [1] | AFREADY | Drives the **AFREADYE** output pin. |
| [0] | ATVALID | Drives the **ATVALIDE** output pin. |

### 4.23.5 TRCITCTRL, Integration Mode Control Register

The TRCITCTRL enables topology detection or integration testing, by putting ETM-M55 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 4-1 ETM-M55 register summary* on page 4-37.

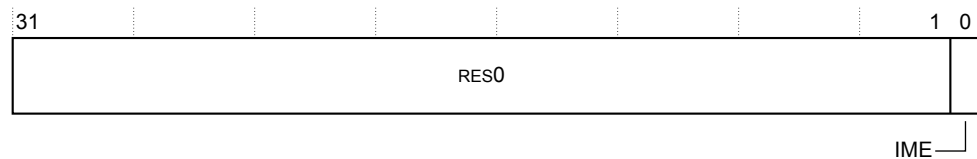The following figure shows the TRCITCTRL bit assignments.



**Figure 4-38 TRCITCTRL bit assignments**

The following table shows the TRCITCTRL bit assignments.

**Table 4-42 TRCITCTRL bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:1] | - | RES0 |
| [0] | IME | Integration mode enable:<br>**0**      ETM-M55 is not in integration mode. This is the reset value.<br>**1**      ETM-M55 is in integration mode. |

## 4.24    TRCCLAIMSET, 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 4-1  ETM-M55 register summary* on page 4-37.

The following figure shows the TRCCLAIMSET bit assignments.



**Figure 4-39  TRCCLAIMSET bit assignments**

The following table shows the TRCCLAIMSET bit assignments.

**Table 4-43  TRCCLAIMSET bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:4] | - | RAZ/SBZ. |
| [3:0] | SET | On reads, each bit is:<br><br>**1**    Claim tag bit is implemented. This value is returned for each of the claim bits 0-3, indicating 4 claim bits are implemented.<br><br>On writes, setting each bit to:<br><br>**0**    Has no effect.<br>**1**    Sets the relevant bit of the claim tag. |

# 4.25 TRCCLAIMCLR, 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 4-1 ETM-M55 register summary* on page 4-37.

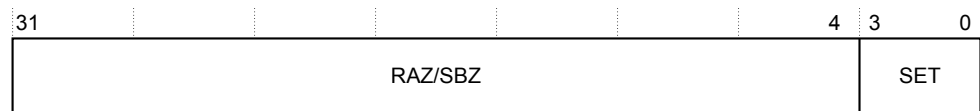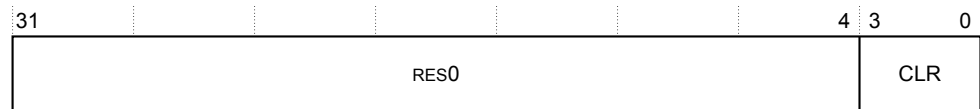The following figure shows the TRCCLAIMCLR bit assignments.



**Figure 4-40 TRCCLAIMCLR bit assignments**

The following table shows the TRCCLAIMCLR bit assignments.

**Table 4-44 TRCCLAIMCLR bit assignments**

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

## 4.26 TRCAUTHSTATUS, 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 4-1 ETM-M55 register summary* on page 4-37.

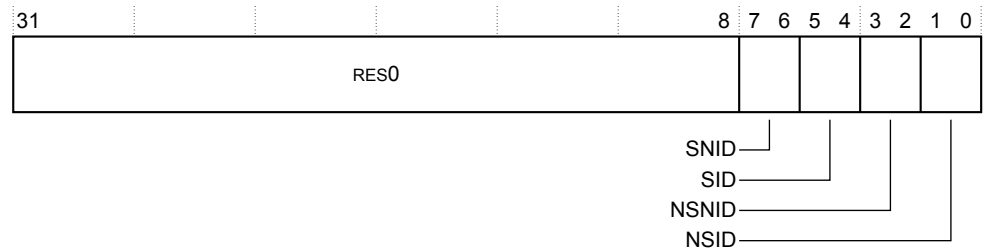The following figure shows the TRCAUTHSTATUS bit assignments.



**Figure 4-41 TRCAUTHSTATUS bit assignments**

The following table shows the TRCAUTHSTATUS bit assignments.

**Table 4-45 TRCAUTHSTATUS bit assignments**

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

---

[c]   SNID bitfield is RAZ when the processor is configured without the Armv8-M Security Extension.

## 4.27    TRCDEVARCH, Device Architecture Register

The TRCDEVARCH identifies ETM-M55 as an ETMv4.5 component.

**Usage constraints**

This register is read-only.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1  ETM-M55 register summary* on page 4-37.

The following figure shows the TRCDEVARCH bit assignments.



**Figure 4-42  TRCDEVARCH bit assignments**

The following table shows the TRCDEVARCH bit assignments.

**Table 4-46  TRCDEVARCH bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:21] | ARCHITECT | Defines the architect of the component:<br><br>`0x23B`                                             Arm. |
| [20] | PRESENT | Indicates the presence of this register:<br><br>`0b1`                      Register is present. |
| [19:16] | REVISION | Architecture revision:<br><br>`0x5`                 Architecture revision 4.5. |
| [15:0] | ARCHID | Architecture ID:<br><br>`0x4A13`                        ETMv4.5 component. |

## 4.28 TRCDEVID, Device ID Register

The TRCDEVID is reserved, RES0.

## 4.29 TRCDEVTYPE, Device Type Register

The TRCDEVTYPE indicates the type of the component.

**Usage constraints**

This register is read-only.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.
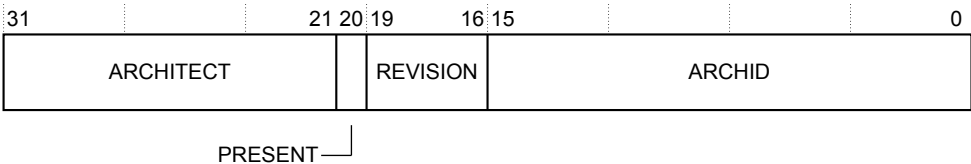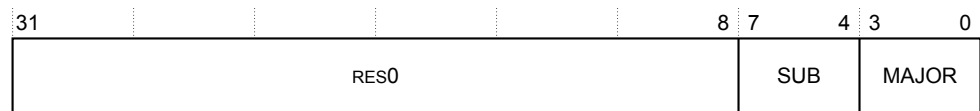
The following figure shows the TRCDEVTYPE bit assignments.



**Figure 4-43 TRCDEVTYPE bit assignments**

The following table shows the TRCDEVTYPE bit assignments.

**Table 4-47 TRCDEVTYPE bit assignments**

| Bits | Name | Function |
|------|------|----------|
| [31:8] | - | RES0. |
| [7:4] | SUB | The subtype of the component: <br> `0b0001` Processor trace. |
| [3:0] | MAJOR | The main type of the component: <br> `0b0011` Trace source. |

## 4.30 TRCPIDR0-7, Peripheral Identification Registers

The TRCPIDR0-7 provides the standard Peripheral ID required by all CoreSight components.

**Usage constraints**

Only bits[7:0] of each register are used. This means that TRCPIDR0-7 define a single 64-bit *Peripheral ID*, as the following figure shows.

**Configurations**

Available in all configurations.

**Attributes**

See the register summary in *Table 4-1 ETM-M55 register summary* on page 4-37.

The following figure shows the mapping between TRCPIDR0-7 and the single 64-bit *Peripheral ID* value.
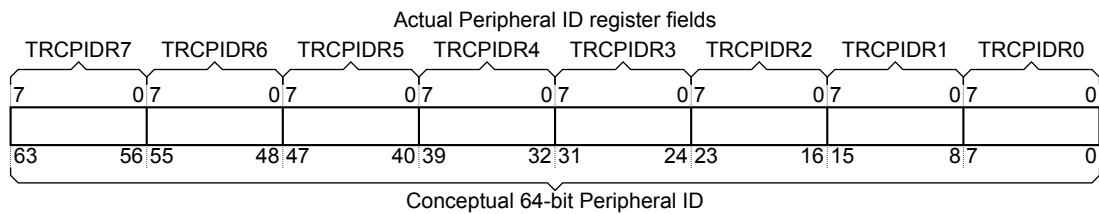


**Figure 4-44 Mapping between TRCPIDR0-7 and the Peripheral ID value**

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



‡ See text for the value of the Revision field

**Figure 4-45 Peripheral ID fields**
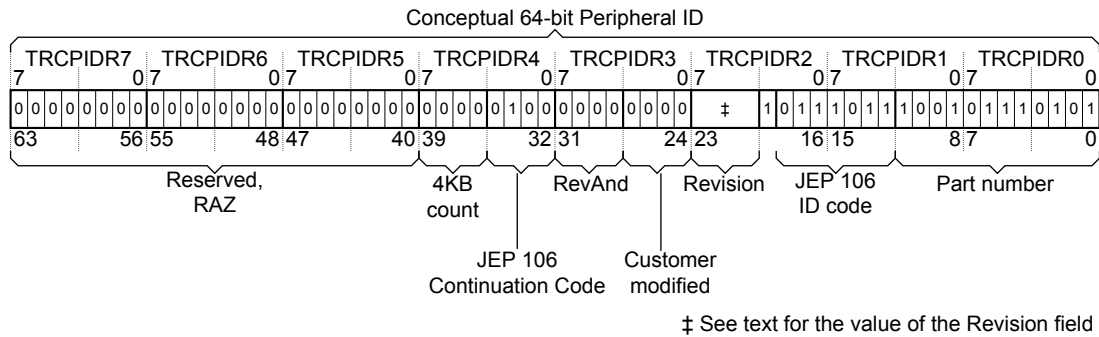
The following table shows the values of the fields when reading this set of registers. The *Arm® Embedded Trace Macrocell Architecture Specification ETMv4* gives more information about many of these fields.

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 4-48  TCRPIDR0-7 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 *4.16.2 TRCIDR1, ID Register 1 on page 4-58*. |
| | | | [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] | 0x22 | Part Number [7:0]. |

## 4.31 TRCCIDR0-3, Component Identification Registers

The TRCCIDR0-3 identifies ETM-M55 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 4-1 ETM-M55 register summary* on page 4-37.

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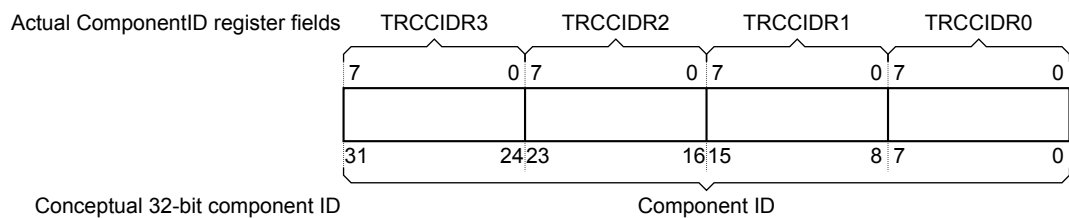


**Figure 4-46 Mapping between TRCCIDR0-3 and the Component ID value**

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 4-49 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]. |

# Appendix A
# **Revisions**

This appendix describes the technical changes between released issues of this book.

It contains the following section:

## A.1 Revisions

The following tables show any significant technical changes between released issues of this book.

**Table A-1 Issue 0000-02**

| Change | Location |
|---|---|
| First Beta release for r0p0 | - |

**Table A-2 Differences between issue 0000-02 and issue 0000-04**

| Change | Location |
|---|---|
| First limited access release for r0p0 | - |
| Low-power override feature corrected to indicate that it is implemented | *1.3 Features* on page 1-15 |
| Example CoreSight system with ETM-M55 updated | *1.1.1 The CoreSight™ debug environment* on page 1-12 |
| Global timestamping functional block description changed. | *Table 2-1 ETM-M55 functional blocks* on page 2-23 |
| Description of external and input and output connection information between the ETM and CTI improved | *2.2 External input and output connections* on page 2-24 |
| Description of parallel instruction execution updated | *2.3.3 Parallel instruction execution* on page 2-30 |
| • Instruction trace resource selection table updated<br>• Trigger event resource selection figure updated | *2.3.6 Resource selection* on page 2-30 and *4.17 TRCRSCTLRn, Resource Selection Registers 2-3* on page 4-67 |
| Removed unnecessary subgrouping of register summary tables. Only single register summary table remains | *4.1 Register summary* on page 4-37 |
| Description of TRCSTALLCTLR.LEVEL field has been updated | *4.7 TRCSTALLCTLR, Stall Control Register* on page 4-46 |
| Description of TRCTRACEIDR has been updated | *4.11 TRCTRACEIDR, Trace ID Register* on page 4-50 |
| Description of TRCSSCSR0.STATUS has been updated | *4.19 TRCSSCSR0, Single-shot Comparator Status Register 0* on page 4-70 |

**Table A-3 Differences between issue 0000-04 and issue 0001-05**

| Change | Location |
|---|---|
| First early access release for r0p1 | - |
| Resource list updated | *2.3.6 Resource selection* on page 2-30 |
| TRCIDR1 reset value updated | • *4.1 Register summary* on page 4-37<br>• *4.16.2 TRCIDR1, ID Register 1* on page 4-58 |
| TRCITIDATAR register description corrected | *4.23.2 TRCITIDATAR, Integration Data Register* on page 4-77 |

**Table A-4  Differences between issue 0001-05 and issue 0002-01**

| Change | Location |
|---|---|
| First release for r0p2 | - |
| TRCIDR1 reset value updated | • *4.1 Register summary* on page 4-37<br>• *4.16.2 TRCIDR1, ID Register 1* on page 4-58 |
| TRCAUXCTLR register reset value corrected | *4.1 Register summary* on page 4-37 |

**Table A-5  Differences between issue 0002-01 and issue 0002-02**

| Change | Location |
|---|---|
| Second release for r0p2 | - |
| PMU events table added | *2.2 External input and output connections* on page 2-24 |
| Supported trace features list updated | *2.3.4 Trace features* on page 2-30 |