



Arm® Embedded Trace Macrocell CoreSight® ETM-R7

Product Revision r0

Software Developers Errata Notice

Non confidential – Released

Software Developers Errata Notice

Copyright © 2013-2019 Arm. All rights reserved.

Non-Confidential Proprietary Notice

This document is protected by copyright and other related rights and the practice or implementation of the information contained in this document may be protected by one or more patents or pending patent applications. No part of this document may be reproduced in any form by any means without the express prior written permission of Arm. **No license, express or implied, by estoppel or otherwise to any intellectual property rights is granted by this document unless specifically stated.**

Your access to the information in this document is conditional upon your acceptance that you will not use or permit others to use the information for the purposes of determining whether implementations infringe any third party patents.

THIS DOCUMENT IS PROVIDED "AS IS". ARM PROVIDES NO REPRESENTATIONS AND NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NON-INFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO THE DOCUMENT. For the avoidance of doubt, Arm makes no representation with respect to, and has undertaken no analysis to identify or understand the scope and content of, patents, copyrights, trade secrets, or other rights.

This document may include technical inaccuracies or typographical errors.

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL ARM BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

This document consists solely of commercial items. You shall be responsible for ensuring that any use, duplication or disclosure of this document complies fully with any relevant export laws and regulations to assure that this document or any portion thereof is not exported, directly or indirectly, in violation of such export laws. Use of the word "partner" in reference to Arm's customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

If any of the provisions contained in these terms conflict with any of the provisions of any click through or signed written agreement covering this document with Arm, then the click through or signed written agreement prevails over and supersedes the conflicting provisions of these terms. This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

The Arm corporate logo and words marked with ® or ™ are registered trademarks or trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. All rights reserved. Other brands and names mentioned in this document may be the trademarks of their respective owners. Please follow Arm's trademark usage guidelines at <http://www.arm.com/company/policies/trademarks>.

Copyright © 2013-2019 Arm Limited (or its affiliates). All rights reserved.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ

Web Address

<http://www.arm.com>

Feedback on content

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

- the document title
- the document number, PJDOC-466751330-11420
- the page numbers to which your comments apply
- a concise explanation of your comments.

General suggestions for additions and improvements are also welcome.

Release Information

Errata are listed in this section if they are new to the document, or marked as “updated” if there has been any change to the erratum text in Chapter 2. Fixed errata are not shown as updated unless the erratum text has changed. The summary table in section 2.2 identifies errata that have been fixed in each product revision.

28 Jun 2019: Changes in Document v2

Page	Status	ID	Cat	Rare	Summary of Erratum
28	New	1076344	CatC		ETM generates incorrectly formatted packet

18 Jun 2013: Changes in Document v1

Page	Status	ID	Cat	Rare	Summary of Erratum
9	New	787176	CatA		The instruction trace stream might be incorrect
9	New	786374	CatB		Exception level and Context ID might not be traced reliably
11	New	787177	CatB		Swap instruction execution might cause incorrect data value
12	New	787181	CatB		Additional atoms committed while cycle count enabled
13	New	790519	CatB		Load from Strongly Ordered Device memory region might not have data value traced
14	New	794920	CatB		Counter self-reload operation is incorrect
15	New	794922	CatB		Register writes are ignored when TRCPRGCTLR.EN is b1 and OS lock is set
16	New	794925	CatB		Claim Tag Set Register (TRCCLAIMSET) behaviour is incorrect
16	New	787178	CatB	Rare	An exception immediately after certain instructions causes incorrect instruction trace
18	New	787319	CatB	Rare	Store Exclusive instruction might cause Single Shot comparators to not fire
19	New	790720	CatB	Rare	ETM might not acknowledge ATB flush request
19	New	780419	CatC		Exception return packet might be delayed
21	New	787179	CatC		ETM acknowledges ATB flush request before all trace has been output
22	New	790572	CatC		Synchronization request might be ignored
23	New	790770	CatC		Periodic synchronization counter reset in low power state
24	New	794919	CatC		The Context ID comparator resource persists between instructions
25	New	794921	CatC		Cycle information in Timestamp packets is incorrect
26	New	794924	CatC		TRCPDSR access is incorrect on memory-mapped access
27	New	796769	CatC		Integration register of ETM-R7 shows incorrect behavior

Contents

CHAPTER 1.	6
INTRODUCTION	6
1.1. Scope of this document	6
1.2. Categorization of errata	6
CHAPTER 2.	7
ERRATA DESCRIPTIONS	7
2.1. Product Revision Status	7
2.2. Revisions Affected	7
2.3. Category A	9
787176: The instruction trace stream might be incorrect.....	9
2.4. Category A (Rare)	9
2.5. Category B	9
786374: Exception level and Context ID might not be traced reliably.....	9
787177: Swap instruction execution might cause incorrect data value	11
787181: Additional atoms committed while cycle count enabled.....	12
790519: Load from Strongly Ordered Device memory region might not have data value traced	13
794920: Counter self-reload operation is incorrect.....	14
794922: Register writes are ignored when TRCPRGCTLR.EN is b1 and OS lock is set	15
794925: Claim Tag Set Register (TRCCLAIMSET) behaviour is incorrect.....	16
2.6. Category B (Rare)	16
787178: An exception immediately after certain instructions causes incorrect instruction trace	16
787319: Store Exclusive instruction might cause Single Shot comparators to not fire	18
790720: ETM might not acknowledge ATB flush request.....	19
2.7. Category C	19
780419: Exception return packet might be delayed.....	19
787179: ETM acknowledges ATB flush request before all trace has been output	21
790572: Synchronization request might be ignored	22
790770: Periodic synchronization counter reset in low power state	23
794919: The Context ID comparator resource persists between instructions	24
794921: Cycle information in Timestamp packets is incorrect.....	25
794924: TRCPDSR access is incorrect on memory-mapped access.....	26
796769: Integration register of ETM-R7 shows incorrect behavior.....	27
1076344: ETM generates incorrectly formatted packet.....	28

Chapter 1.

Introduction

This chapter introduces the errata notice for the CoreSight ETM-R7.

1.1. Scope of this document

This document describes errata categorized by level of severity. Each description includes:

- the current status of the defect
- where the implementation deviates from the specification and the conditions under which erroneous behavior occurs
- the implications of the erratum with respect to typical applications
- the application and limitations of a 'work-around' where possible

This document describes errata that may impact anyone who is developing software that will run on implementations of this ARM product.

1.2. Categorization of errata

Errata recorded in this document are split into the following levels of severity:

Table 1 **Categorization of errata**

Errata Type	Definition
Category A	A critical error. No workaround is available or workarounds are impactful. The error is likely to be common for many systems and applications.
Category A(rare)	A critical error. No workaround is available or workarounds are impactful. The error is likely to be rare for most systems and applications. Rare is determined by analysis, verification and usage.
Category B	A significant error or a critical error with an acceptable workaround. The error is likely to be common for many systems and applications.
Category B(rare)	A significant error or a critical error with an acceptable workaround. The error is likely to be rare for most systems and applications. Rare is determined by analysis, verification and usage.
Category C	A minor error.

Chapter 2.

Errata Descriptions

2.1. Product Revision Status

The *rn* identifier indicates the revision status of the product described in this book, where:

- rn** Identifies the major revision of the product.
- pn** Identifies the minor revision or modification status of the product.

2.2. Revisions Affected

Table 2 below lists the product revisions affected by each erratum. A cell marked with **X** indicates that the erratum affects the revision shown at the top of that column.

This document includes errata that affect revision r0 only.

Refer to the reference material supplied with your product to identify the revision of the IP.

Table 2 **Revisions Affected**

ID	Cat	Rare	Summary of Erratum	r0p0	r0p1
787176	CatA		The instruction trace stream might be incorrect	X	
794925	CatB		Claim Tag Set Register (TRCCLAIMSET) behaviour is incorrect	X	
794922	CatB		Register writes are ignored when TRCPRGCTLR.EN is b1 and OS lock is set	X	
794920	CatB		Counter self-reload operation is incorrect	X	
790519	CatB		Load from Strongly Ordered Device memory region might not have data value traced	X	
787181	CatB		Additional atoms committed while cycle count enabled	X	
787177	CatB		Swap instruction execution might cause incorrect data value	X	
786374	CatB		Exception level and Context ID might not be traced reliably	X	
790720	CatB	Rare	ETM might not acknowledge ATB flush request	X	
787319	CatB	Rare	Store Exclusive instruction might cause Single Shot comparators to not fire	X	
787178	CatB	Rare	An exception immediately after certain instructions causes incorrect instruction trace	X	
1076344	CatC		ETM generates incorrectly formatted packet	X	X
796769	CatC		Integration register of ETM-R7 shows incorrect behavior	X	X
794924	CatC		TRCPDSR access is incorrect on memory-mapped access	X	
794921	CatC		Cycle information in Timestamp packets is incorrect	X	
794919	CatC		The Context ID comparator resource persists between instructions	X	
790770	CatC		Periodic synchronization counter reset in low power state	X	
790572	CatC		Synchronization request might be ignored	X	

ID	Cat	Rare	Summary of Erratum	r0p0	r0p1
787179	CatC		ETM acknowledges ATB flush request before all trace has been output	X	
780419	CatC		Exception return packet might be delayed	X	

2.3. Category A

787176: The instruction trace stream might be incorrect

Category A

Products Affected: CoreSight ETM-R7.

Present in: r0p0

Description

The ETM might indicate either:

- That one or more instructions have been architecturally executed by the processor when they have not.
- That one or more instructions have not been architecturally executed by the processor when they have.

This erratum is detectable at periodic synchronization points in the trace stream.

Conditions

This erratum occurs when there is misspeculation by the processor. Specific timing conditions must also occur in the processor and in the ETM.

Implications

The ETM might commit one or more atoms representing instructions that are not architecturally executed by the processor. Similarly, the ETM might cancel one or more atoms representing instructions that are architecturally executed by the processor. ViewInst start/stop and Single-Shot are also affected and might become incorrect.

Workaround

There is no workaround for this erratum.

The erratum is detectable at a periodic synchronization point, when the speculation depth of the decompressor is inconsistent with the speculation depth of the Trace Info element. The speculation depth of the Trace Info element is correct. A decompressor might use the speculation depth of the Trace Info element to recover from previous incorrect trace.

After the Trace Info element, trace is correct and decompressible.

2.4. Category A (Rare)

There are no errata in this category

2.5. Category B

786374: Exception level and Context ID might not be traced reliably

Category B

Products Affected: CoreSight ETM-R7.

Present in: r0p0

Description

When instruction trace is enabled, the ETM traces exception level information and can be configured to trace Context ID.

Because of this erratum, the ETM might not trace Context changes correctly and instructions might therefore be traced with the wrong exception level or Context ID.

Instruction and data trace are otherwise correct and fully decompressible. This erratum does not impact Address Comparators, Context ID Comparators, or ViewInst filtering based upon exception level.

Conditions

Incorrect exception level information might occur when instruction trace is enabled and the exception level changes. Incorrect Context ID trace might occur when instruction trace is enabled, context trace is enabled and the Context ID changes.

Specific timing conditions must also be present inside the ETM.

Implications

The ETM might trace a context change earlier or later than expected. As a result, exception level and Context ID information might not be traced reliably.

This erratum does not impact resources. Context ID Comparators and Address Comparators behave correctly. ViewInst filtering based upon exception level behaves correctly.

Workaround

There is no workaround for this erratum.

787177: Swap instruction execution might cause incorrect data value**Category B****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

ETM data value trace is enabled using the TRCCONFIGR.DV register field. When data value trace is enabled, data addresses and data values are traced as they occur.

In rare conditions, when data value trace is enabled, the execution of a swap instruction, SWP or SWPB, can cause incorrect data value trace, meaning data values of the swap instruction are incorrect. Data values of preceding data instructions might also be incorrect or missing. At most, two previous data transfers might also be affected. All other data address and data value pairs are not affected. The data trace stream remains completely decompressible.

The instruction trace stream is not affected.

This erratum might occur on SWP and SWPB instructions.

Conditions

This erratum might occur if all of the following conditions exist:

- Data value tracing is enabled in the ETM. TRCCONFIGR.DV is b1.
- A SWP or SWPB instruction is executed by the processor.
- Specific timing conditions occur in the processor and ETM.

Implications

The trace key of a preceding instruction data value is incorrect. As a result, at most one previously-traced data instruction is impacted, resulting in incorrect or missing data values for this instruction. The data instruction impacted by this erratum is not necessarily the data instruction immediately preceding the swap.

If the swap data is traced by the ETM, data values of the swap instruction are also affected as follows:

- The reported load value of the swap instruction is the store value.
- The reported store value of the swap instruction is incorrect.

For this erratum to occur, the swap instruction only needs to be executed by the processor, and not necessarily traced by the ETM.

Workaround

There is no workaround for this erratum. The use of swap instructions is deprecated by ARM.

787181: Additional atoms committed while cycle count enabled**Category B****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

When enabled in the ETM, cycle count packets are inserted in the instruction trace stream to provide information about the passage of processor clock cycles relative to the execution of instructions in the stream.

Under rare conditions, when cycle counting is enabled, the ETM will indicate that the processor has executed instructions that have not yet been executed.

Instruction trace remains fully decompressible.

Conditions

This erratum occurs if all of the following conditions exist:

- Cycle counting is enabled. TRCCONFIGR.CCI is programmed to b1.
- A Cycle Count Format 2 or Cycle Count Format 3 packet is driven out of the ETM.
- Specific timing conditions occur in the ETM.

Implications

The Cycle Count Format 2 or Format 3 packet indicates one P0 Element is committed when it should not have been. Depending upon the subsequent trace, this erratum has one of two possible implications:

- If no Cancel Elements are issued for the remainder of the trace session, at the end of the session, one P0 Element is committed that should not be. This incorrectly indicates that the processor has executed more instructions than it has.
- If at least one Cancel Element is issued before the end of the session, trace will indicate the cancellation of one P0 Element that was incorrectly committed.

Workaround

If no Cancel Elements are issued for the remainder of the trace session, there is no work-around. However, committed trace that does not correspond to instruction execution is limited to P0 Elements occurring after the most recent Cancel Element.

If a Cancel Element indicates a previously-committed P0 Element has been cancelled, a decompressor should apply the cancel to the previously-committed P0 Element.

This erratum might occur cumulatively. If several Cycle Count Format 2 or Format 3 packets commit an incorrect number of P0 Elements, a subsequent Cancel Element will cancel more than one previously-committed P0 Elements.

This erratum does not occur if cycle counting is disabled.

790519: Load from Strongly Ordered Device memory region might not have data value traced**Category B****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

ETM data value trace is enabled using the TRCCONFIGR.DV register field. When data value trace is enabled, data addresses and data values are traced as they occur. ETM comparators can also be used to compare data addresses and data values, or assert when data is executed by the processor.

In certain conditions, when loads are performed in a Strongly Ordered or Device memory region, if an FIQ or IRQ follows, data values of the loads might not be traced. Data value comparisons are also affected and might not assert as expected. Single-Shot Comparators are also affected and might not assert as expected.

All other data address and data value pairs are not affected. The data trace stream remains completely decompressible. The instruction trace stream is not affected.

Conditions

The erratum can only occur if all of the following conditions are met:

- One or more load instructions are executed, where the data is loaded from Strongly Ordered or Device memory regions.
- An FIQ or IRQ exception occurs and data values of the load instructions have not been returned from memory.

Specific conditions must also occur inside the processor.

Implications

If data value trace is enabled, data values of the load instructions are not traced. If a data value comparison is programmed to assert on the data values, the comparison will not assert as expected. If a Single-Shot Comparator is programmed to assert on data addresses of the loads, the comparator will not assert as expected. If a Single-Shot Comparator is programmed to assert on data addresses and data values of the loads, the comparator will not assert as expected.

This erratum has no impact on other instructions or data values. Data trace remains decompressible.

Workaround

There is no workaround for this erratum.

794920: Counter self-reload operation is incorrect**Category B****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

The ETM filtering resources include counters that can be configured to count down and reload with a pre-configured value on reaching zero. The counters have a resource output that is active when the counter is at zero.

Because of this erratum, the counter at zero resource is active on every cycle when the counter value is zero, even when the counter is configured in self-reload mode. The correct operation in self-reload mode is for the resource to only be active for the cycle when the count value is zero and the count enabling event is active.

Chaining of counters to provide a single larger counter is not affected.

Conditions

This erratum occurs if all of the following conditions exist:

- A counter is configured in self-reload mode.
- The counter at zero resource from this counter is used.
- The decrement event can be not active when the counter is at zero.

Implications

The counter at zero resource will be active on every cycle when the counter value is zero. This resource will therefore be active for more cycles than expected.

If the counter is programmed with a count value of zero, the counter at zero resource starts active even when the counter is configured for self-reload mode.

Workaround

There is no workaround for this erratum.

794922: Register writes are ignored when TRCPRGCTLR.EN is b1 and OS lock is set**Category B****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

The ETM is enabled by setting the enable bit in the Programming Control Register. The OS lock is set by writing the OS lock key in the OS Lock Access Register. When the ETM is enabled and the OS lock is set, register writes to the ETM Trace registers are ignored.

The correct operation is that they should be allowed.

Conditions

This erratum occurs only if all the following conditions are met:

- The ETM is enabled. TRCPRGCTLR.EN is set to b1.
- The OS lock bit is set. TRCOSLAR.OSLK is set to b1.

Implications

Under these conditions, register writes to the ETM are ignored. Register reads behave correctly.

This erratum impacts the save/restore sequence of the ETM.

Workaround

When issuing a store sequence, TRCPRGCTLR.EN must be cleared at the start of the store sequence. When issuing a restore sequence, TRCPRGCTLR.EN must be restored after all other registers have been restored.

794925: Claim Tag Set Register (TRCCLAIMSET) behaviour is incorrect**Category B****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

If an ETM is shared between multiple software and external debugger devices, claim tags can be used to coordinate application and debugger access to the ETM.

To facilitate coordination, writing zero to a bit in the Claim Tag Set Register should have no effect. However, because of this erratum, writing zero to a bit in TRCCLAIMSET clears the respective claim tag bit.

The claim tags have no effect on the operation of trace unit.

Conditions

This erratum occurs when the TRCCLAIMSET register is written.

Implications

Because writing zero to a claim tag bit clears the bit, claim tags cannot be used to coordinate ETM access between multiple sources.

Workaround

There is no workaround for this erratum.

2.6. Category B (Rare)

787178: An exception immediately after certain instructions causes incorrect instruction trace**Category B Rare****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

When exceptions are traced, the exception return address is traced and speculation is resolved.

In rare conditions, if an exception occurs immediately following certain instructions, the ETM might cancel atoms representing instructions that are architecturally executed by the processor. The preferred exception return address is correct. Although trace is incomplete, the instruction trace stream is decompressible following the exception.

Conditions

This erratum can occur only if all of the following conditions are met:

- One of the following instructions is executed:
 - An instruction that updates the CPSR from the SPSR
 - WFI/WFE
 - MCR/MRC/MCRR/MRRC
 - CPS/SETEND/CLKREX/DSB/RFE
- An exception occurs before execution of this instruction completes.
- Specific timing conditions are present when the exception occurs, including unresolved speculation.

Implications

The ETM might incorrectly cancel one or more atoms representing instructions that are architecturally executed by the processor. This might result in a gap between the address of the last committed instruction and the address of the instruction immediately following the exception. ViewInst start/stop and single-shot are also affected and might become incorrect.

The instruction trace stream after the Exception element is correct and decompressible

Workaround

This erratum is not reliably detectable. If a preferred exception return address is the same as one of the above-listed instructions, this erratum might cause a gap between the last committed instruction and the preferred exception return address.

There is no workaround for this erratum.

787319: Store Exclusive instruction might cause Single Shot comparators to not fire**Category B Rare****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

A Single Shot comparator can be configured to fire when a data address or range of data addresses are accessed by the processor.

In rare conditions, when a Single Shot comparator is configured to fire on a data address comparison and an instruction performs a data access that should fire the Single Shot comparator, a subsequent Store Exclusive instruction might cause the Single Shot comparator to not fire as expected.

This erratum only affects a Single Shot comparison that should be triggered by the instruction before the Store Exclusive instruction. Any Single Shot comparison triggered by the store exclusive instruction is not affected.

The trace stream remains completely decompressible.

Conditions

The erratum can only occur if all of the following conditions are met:

- A Single Shot comparator is configured to fire on a single data address or a range of data addresses, with or without data value comparison.
- An instruction accesses data that should cause the Single Shot comparator to fire.
- A Store Exclusive instruction follows the instruction that should fire the comparator.

Specific timing conditions must also be present inside the ETM.

Implications

The Single Shot comparator does not fire as expected.

This erratum is independent of whether a single address (SAC) or an address range (ARC) is used as the input to the Single Shot comparator.

Workaround

There is no workaround for this erratum.

790720: ETM might not acknowledge ATB flush request**Category B Rare****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

When the trace capture infrastructure requests an ATB flush, the ETM should acknowledge the flush request when all trace in the system at the time of request has been output from the ETM.

In rare conditions, a flush request is never acknowledged. Instruction and data trace are not affected by this erratum and remain completely decompressible.

Conditions

This erratum occurs only if all of the following conditions are met:

- The processor enters low power state by executing a WFI or WFE instruction. The processor exits low power state before the drain completes.
- An ETM overflow occurs.
- An ATB flush request occurs before the Overflow packet has been emitted.

Specific timing conditions must also be present inside the ETM.

Implications

Under these conditions, the flush request might not be acknowledged. The flush request is acknowledged only when the ETM has been disabled or when the processor subsequently enters low power state by executing a WFI or WFE instruction.

This erratum can occur on both the instruction and data ATB interfaces.

If the ETM is connected as part of a CoreSight Trace Funnel and a flush request is issued on the funnel, because of this erratum, the funnel might not acknowledge the flush request.

Workaround

There is no reliable detection mechanism for this erratum. If a flush request is not acknowledged and an overflow has occurred in the trace stream, this erratum might be the cause.

A suitable workaround for this erratum is to override the ETM low power state behavior by assigning TRCEVENTCTL1R.LPOVERRIDE a value of b1. This might increase power consumption.

2.7. Category C

780419: Exception return packet might be delayed**Category C****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

The ETMv4 architecture requires an exception return (ERET) packet to be traced after every exception return instruction. This is to indicate when the instruction stream exits an exception handler without the need to completely decompress the trace stream.

In rare conditions the exception return packet cannot be traced in the same cycle as the exception return and is then delayed until after the next P0 element.

The trace stream is completely decompressible and all the required packets are present.

Conditions

- 1) Context ID tracing must be enabled.
- 2) A change in exception level or context ID has been traced.
- 3) One or more instructions are executed.
- 4) An exception return instruction is executed which does not change the CPU state.
- 5) The next instruction is a P0 element.

The ERET packet is traced after step 5. It should be traced after step 4.

Implications

The position of the exception return instruction might not be exactly inferred from the position of the exception return packet in the trace stream. The exception return instruction can be determined from the instruction opcode. The points of entry and exit from exception handlers can still be determined without the need to completely decompress the trace stream.

Workaround

There is no workaround for this erratum.

787179: ETM acknowledges ATB flush request before all trace has been output**Category C****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

When an ATB flush is requested by the trace capture infrastructure, the ETM should acknowledge the flush request only when all trace in the system at the time of request has been output from the ETM.

The ETM is capable of generating ATB triggers independently of flush requests. When configured appropriately, the ETM will drive an ATB trigger when a certain event fires in the ETM resources.

In rare conditions, if an ATB trigger coincides with a flush acknowledge on the ATB interface, the ETM flush acknowledge might be presented before all data has been drained from the ETM.

This erratum is present on the instruction ATB interface and the data ATB interface. When the erratum occurs, there is an upper bound on the number of transactions that are still buffered in the ETM when the flush is acknowledged early.

Conditions

This erratum can only occur if all of the following conditions are met:

- TRCEVENTCTL0R.EVENT0 is programmed to generate an event. TRCEVENTCTL1R.ATB is programmed to b1, so that resource event0 asserting is followed by an ATB trigger.
- An ATB flush is requested.
- Resource event 0 fires, generating an ATB trigger.
- Specific timing conditions are met in the ETM, including the flush acknowledge being driven at exactly the same time as the ATB trigger.

Implications

The ETM acknowledges the ATB flush request before all trace has been output.

If the erratum occurs on the instruction ATB interface, at most 4 instruction bytes are still buffered in the ETM when the flush is incorrectly acknowledged. This represents at most 96 Atom elements.

If the erratum occurs on the data ATB interface, at most 8 data bytes are still buffered in the ETM when the flush is incorrectly acknowledged. This represents at most 64 P1 Data Address elements or 16 P2 Data Value elements, or a combination of fewer P1 Data Address elements and P2 Data Value elements.

Workaround

For flush to work precisely, the workaround is to disable ATB trigger generation by programming TRCEVENTCTL1R.ATB to 0. Otherwise, there is no workaround for this erratum.

790572: Synchronization request might be ignored**Category C****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

Synchronization requests can be issued by several sources, including a periodic counter in the trace unit, inputs into the trace unit, or when buffer overflow occurs.

In certain conditions, a synchronization request issued by a periodic counter or trace unit input might be ignored.

Conditions

This erratum can only occur if all the following conditions are met:

- Event trace is enabled. TRCEVENTCTL1R.INSTEN is b1 or TRCEVENTCTL1R.DATAEN is b1.
- A new trace session is started or a synchronization request is issued.
- Instruction or Data Events are traced and no atoms are traced (ViewInst is inactive).
- A subsequent synchronization request is issued because of the periodic counter or trace unit input.

Implications

The second synchronization request is ignored. This erratum does not occur if the second synchronization is caused by buffer overflow or a new trace session.

If trace is captured in a circular buffer and only event trace is captured, then there might be no synchronization events in the buffer.

Workaround

If trace is captured in a circular buffer, a suitable workaround for this erratum is to ensure that sufficient instructions are traced to allow synchronization to occur, which might be achieved by using a counter to periodically enable and disable trace. This might be achieved with the following register sequence:

Configure Counter 0 to always reload when it reaches 0. Set Counter 0 decrement event to TRUE.

- TRCCNTCTLR0.CNTCHAIN = 0b0
- TRCCNTCTLR0.RLDSELF = 0b1
- TRCCNTCTLR0.RLDEVENT = 0b00000000
- TRCCNTCTLR0.CNTEVENT = 0b00000001

Configure Counter 0 Reload Value of 0x1000.

- TRCCNTRLDR0.VALUE = 0x1000

Configure Resource Selection Control Register 2 for Counter 0.

- TRCRSCTLR2.PAIRINV = 0b0
- TRCRSCTLR2.INV = 0b0
- TRCRSCTLR2.GROUP = 0b0010
- TRCRSCTLR2.SELECT = 0x0001

Enable ViewInst when the Resource Selector 2 asserts.

- TRCVICTLR.EVENT = 0b00000010

790770: Periodic synchronization counter reset in low power state**Category C****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

Synchronization requests can be issued by several sources, including a periodic counter in the trace unit, inputs into the trace unit, or when buffer overflow occurs.

When the processor enters low power state because of a WFI or WFE instruction, the periodic synchronization counter is reset. As a result, a periodic synchronization request might not be issued when the processor exits low power state.

Instruction and data trace are not affected by this erratum and remain completely decompressible.

Conditions

This erratum only occurs if all of the following conditions are met:

- The periodic synchronization counter is configured to issue periodic synchronization requests. This is achieved by setting TRCSYNCPR.PERIOD to a non-zero value.
- The processor enters low power state by executing a WFI or WFE instruction.

Implications

When the processor exits low power state, the periodic synchronization counter is reset. As a result, a periodic synchronization request might be dropped and synchronization might not occur as frequently as expected.

If trace is captured in a circular buffer, then there might be no synchronization events in the buffer and trace might not be decompressible.

Workaround

If trace is captured in a circular buffer, a suitable workaround for this erratum is to ensure that sufficient synchronization requests are issued by the periodic synchronization counter. This might be achieved by decreasing the TRCSYNCPR.PERIOD value such that synchronization requests from the periodic synchronization counter occur more frequently than entry into low power state. This might increase trace bandwidth.

This erratum can be avoided by overriding the ETM low power state behavior by assigning TRCEVENTCTL1R.LPOVERRIDE a value of b1. This might increase power consumption.

794919: The Context ID comparator resource persists between instructions**Category C****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

The ETM filtering resources include a Context ID comparator which can be combined with instruction comparators or used directly to indicate the execution of instructions that match the specified value.

When used directly, the Context ID comparator resource remains set continuously from the execution of a matching instruction until the cycle when another instruction is executed. The correct operation is for the Context ID comparator to only be set on the cycle an instruction matches.

Conditions

Context ID comparator is used as a resource.

Implications

Any event which uses the Context ID comparator will indicate cycles that match the specified context rather than instructions that match this context. The Context ID comparator will therefore be active for more cycles than expected.

Address comparator resources, when they are sensitive to Context ID, are not affected and operate as specified by the ETMv4 architecture.

Workaround

A suitable workaround exists for this erratum. Instead of using a Context ID comparator directly, the correct behaviour can be reconstructed by using an Address Range Comparator that is sensitive to the Context ID comparator. This can be achieved by configuring the Address Range Comparator as follows:

- TRCACATR<n>.DATAMATCH = b00 ; Do not match on data value
- TRCACATR<n>.EXLEVEL_S = b0000 ; Match on all secure exception levels
- TRCACATR<n>.CONTEXT = b000 ; Select Context ID comparator 0
- TRCACATR<n>.CONTEXTTYPE = b01 ; Use the Context ID comparator
- TRCACATR<n>.TYPE = b00 ; Perform instruction address comparison

Address Comparator Value Registers can then be configured to compare the whole instruction address space. This can be achieved by configuring the Address Comparator Value Registers as follows:

- TRCACVR<2n>.ADDRESS = 0x00000000
- TRCACVR<2n+1>.ADDRESS = 0xffffffff

By using the Address Range Comparator instead of the Context ID comparator, an event will see the expected behaviour.

794921: Cycle information in Timestamp packets is incorrect**Category C****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

The ETM can be configured to trace cycle timing information in the trace stream. Global timestamps can also be included in the trace stream.

When cycle counting is enabled, incremental cycle counts are provided which relate to instruction execution, and specifically to the resolution of speculation.

When timestamping and cycle counting are both enabled, every Timestamp packet includes a cycle count value. This cycle count value indicates the position of the trace element that was timestamped, but is not necessary to reconstruct the cycle count information.

Because of this erratum, the cycle count values provided in Timestamp packets might be unreliable and should not be used.

Conditions

This erratum occurs if all of the following conditions exist:

- Global timestamping is enabled.
- Cycle counting trace is enabled.
- Instruction or event tracing is active.

Implications

The cycle count information for Commit packets is correct, and can be fully reconstructed.

The timestamp values in Timestamp packets cannot be precisely correlated with the commit stream, and it is not possible to determine when the cycle count information in a Timestamp packet is correct.

Workaround

There is no workaround for this erratum.

794924: TRCPDSR access is incorrect on memory-mapped access**Category C****Products Affected: CoreSight ETM-R7.****Present in: r0p0****Description**

The Power Down Status Register indicates the power down status of the trace unit.

This register should be visible to memory-mapped and external access. However, because of this erratum this register is only visible to external access.

This register is a status register and has no impact on operation of the trace unit.

Conditions

This erratum occurs when TRCPDSR is read using the memory-mapped interface.

Implications

When the Power Down Status Register is read from the memory-mapped interface, the register will always return a read value of 0.

Under certain conditions, reading the Power Down Status Register should clear the STICKYPD register field. However, reading this register from the memory-mapped interface has no impact on the STICKYPD register field.

This erratum does not impact accesses via the external debugger interface.

Workaround

There is no workaround for this erratum.

796769: Integration register of ETM-R7 shows incorrect behavior**Category C****Products Affected: CoreSight ETM-R7.****Present in: r0p0, r0p1****Description**

The ETM macrocell supports an integration test mode which permits direct control of some top level pins to provide simple testing of connections between components.

The TRCITATBIDR register is intended to provide control of the ATIDMlx and ATIDMDx top level pins when integration mode is enabled. If this erratum occurs, the module-level clock gating within the ETM prevents any update on these pins until a subsequent APB access is performed.

Conditions

This erratum occurs if all of the following conditions exist:

- Integration mode is enabled (TRCITCTRL.IME = 1)
- TRCITATBIDR is written

Implications

Integration tests might observe the side effect of writing to TRCITATBIDR and then checking a value in the connected ATB peripheral. Because of this erratum, no change is observed. This erratum does not affect the sequence recommended for topology detection.

The use of TRCITATBIDR is limited to testing during the development of a SoC.

Workaround

A work-around is to perform a second access to the APB (read or write) to any ETM register after any write to the TRCITATBIDR.

1076344: ETM generates incorrectly formatted packet**Category C****Products Affected: CoreSight ETM-R7.****Present in: r0p0, r0p1****Description**

The ETM for Cortex-R7 can be configured to trace condition code information for all instructions which are traced. When this trace is enabled, some internal timing conditions might result in a Conditional Instruction Format 3 packet being generated, where the payload has both Z and NUM fields set to zero. This payload is not defined in the ETMv4 trace architecture.

Configurations affected

All configurations including an ETM are affected.

Conditions

The ETM is enabled and tracing.

Conditional non-branch instructions are traced (TRCCONFIGR.COND is non-zero)

Implications

A proportion of conditional instructions will result in the generation of 2 additional bytes of trace which are not required to fully decompress the trace. These bytes correspond to a reserved encoding in the ETM architecture but are consistent with the Conditional Instruction Format 3 header and a single payload byte. If this packet is ignored, then no other aspects of the trace are affected. The trace bandwidth impact should be small.

Workaround

When a packet consisting of bytes 0x6D and 0x00 is received on the instruction trace stream, the decompressor should ignore these two bytes and process the next byte as a new header.