ARM ETM
CoreSight™ ETM-R4 (TM930)
Errata Notice

This document contains all errata known at the date of issue in supported releases up to and including revision r2p1 of CoreSight ETM-R4 (TM930) product.

Proprietary notice

Document confidentiality status

This document is Non Confidential.

Web address

http://www.arm.com/

Feedback on the product

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

- The product name
- A concise explanation of your comments.

Feedback on this document

If you have any comments on about this document, please send email to mailto:support-cores@arm.com giving:

- The document title
- The documents number
- The page number(s) to which your comments refer
- A concise explanation of your comments

General suggestion for additions and improvements are also welcome.

Contents

Introduction

Scope

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

- a unique defect tracking identifier
- 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

Categorisation of Errata

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

Category 1	Behavior that is impossible to work around and that severely restricts the use of the product in all, or the majority of applications, rendering the device unusable.
Category 2	Behavior that contravenes the specified behavior and that might limit or severely impair the intended use of specified features, but does not render the product unusable in all or the majority of applications.
Category 3	Behavior that was not the originally intended behavior but should not cause any problems in applications.
Implementation	Errata that are of particular interest to those implementing the product and that have no software implications

Change Control

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 text of the erratum Description, Conditions, Implications or Workaround. Fixed errata are not shown as updated, unless the erratum text has changed. The Summary Table identifies errata that have been fixed in each product revision.

11 Apr 2011: Changes in Document v7

No new or updated errata in this document version.

Refer to the Summary Table for errata fixes in the latest product revision.

12 Nov 2010: Changes in Document v6

Page	Status	ID	Cat	Summary
25	New	728501	Cat 3	ETM-R4 fails to trace a VnT packet for the second half of a SWP instruction

11 Sep 2008: Changes in Document v5

No new errata in this document revision

27 Jun 2008: Changes in Document v4

Page	Status	ID	Cat	Summary
24	New	550515	Cat 3	Consecutive flushes might not be acknowledged

21 Aug 2007: Changes in Document v3

Page	Status	ID	Cat	Summary
15	New	450964	Cat 2	Trigger might not occur
21	New	445913	Cat 3	Address Range comparator is not cleared by a coprocessor access
22	New	445931	Cat 3	PCLKENDBG not used in claim tag register
23	New	450966	Cat 3	Multiple trigger requests might occur

21 Sep 2006: Changes in Document v2

Page	Status	ID	Cat	Summary
11	New	400022	Cat 2	Accesses to ATCLK registers can cause APB interface to lockup
10	New	379388	Cat 2	De-assertion of NIDEN and DBGEN might cause incorrect trace
13	New	400025	Cat 2	AFREADYM signal might not be asserted
16	New	379031	Cat 3	ETM address range comparator at bottom of memory does not match on unaligned wraparound access
18	New	379386	Cat 3	Lock Access can be modified when PADDRDBG31 is HIGH
19	New	401329	Cat 3	Power Down Status Register (PDSR) reads as zero
20	New	402614	Cat 3	Cycle count is up to 7 cycles greater than it should be on debug exit

20 Jan 2006: Changes in Document v1

There are no errata contained in this version

Errata Summary Table

The errata associated with this product affect product versions as below.

A cell shown thus **X** indicates that the defect affects the revision shown at the top of that column.

ID	Cat	Summary of Erratum	r0p0	r1p0	r2p0	r2p1
379388	Cat 2	De-assertion of NIDEN and DBGEN might cause incorrect trace	X			
400022	Cat 2	Accesses to ATCLK registers can cause APB interface to lockup	X			
400025	Cat 2	AFREADYM signal might not be asserted	X			
450964	Cat 2	Trigger might not occur	X			
379031	Cat 3	ETM address range comparator at bottom of memory does not match on unaligned wraparound access	X			
379386	Cat 3	Lock Access can be modified when PADDRDBG31 is HIGH	X			
401329	Cat 3	Power Down Status Register (PDSR) reads as zero	X			
402614	Cat 3	Cycle count is up to 7 cycles greater than it should be on debug exit	X			
445913	Cat 3	Address Range comparator is not cleared by a coprocessor access	X			
445931	Cat 3	PCLKENDBG not used in claim tag register	X			
450966	Cat 3	Multiple trigger requests might occur	X			
550515	Cat 3	Consecutive flushes might not be acknowledged	X	X		
728501	Cat 3	ETM-R4 fails to trace a VnT packet for the second half of a SWP instruction	X	X	X	

Errata - Category 1

There are no Errata in this Category

Errata - Category 2

379388: De-assertion of NIDEN and DBGEN might cause incorrect trace

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 2, Present in: r0p0, Fixed in r1p0.

Description

Two inputs signals, NIDEN and DBGEN, are provided on CoreSight ETM-R4 as global control signals to allow the CoreSight ETM-R4 to operate. If both of these signals are LOW, then the CoreSight ETM-R4 should stop tracing and output all trace currently in the FIFO. If either of the signals is subsequently driven HIGH, the CoreSight ETM-R4 should restart tracing at the following instruction boundary.

If this erratum occurs, the ETM FIFO does not empty when these signals are driven LOW. When one of the signals is driven HIGH, the data remaining in the FIFO is output, but the ETM might not restart tracing correctly and might output incorrect trace. Packet boundary synchronisation is maintained.

Conditions

The following operations must occur in the sequence defined:

1. The CoreSight ETM-R4 is enabled and generating trace
2. NIDEN and DBGEN are driven LOW
3. NIDEN or DBGEN is driven HIGH

Implications

The trace data is incorrect until the next I-Sync packet or indirect branch packet.

It is not expected that the signals NIDEN and DBGEN will be dynamically changed during tracing since the normal usage model for these signals is to permanently disable tracing on a device.

If these signals are not dynamically changed during tracing, this erratum does not occur.

Workaround

There is no workaround for this erratum.

Trace synchronisation can be regained at the next indirect branch packet or I-Sync packet.

400022: Accesses to ATCLK registers can cause APB interface to lockup

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 2, Present in: r0p0, Fixed in r1p0.

Description

The CoreSight ETMR4 performs internal clock gating to decrease power consumption. Many of the internal registers are clock gated when any of the following conditions exist:

1. The ETMPWRDOWN (bit [0] of the ETM Control register, 0x000) is HIGH
2. The input signals NIDEN and DBGEN are both LOW

When the internal registers are clock gated, accesses to the programmer's model registers in the ATCLK domain cause the output signal PREADYDBG to remain LOW, thereby causing the APB interface to lock up and preventing further accesses to the ETM and other devices on the same APB bus until the ETM is reset.

Conditions

1. The ETMPWRDOWN (bit [0] of the ETM Control register, 0x000) is HIGH, OR the input signals NIDEN and DBGEN are both LOW
2. One of the following registers is accessed:

- CoreSight Trace ID register (register 0x080, offset 0x200)
- Integration Register ITATBCTR0 (register 0x3BE, offset 0xEF8)
- Integration Register ITATBCTR1 (register 0x3BD, offset 0xEF4)
- Integration Register ITATBCTR2 (register 0x3BC, offset 0xEF0)
- Integration Register ITATBDATA0 (register 0x3BB, offset 0xEEC)
- Integration Register ITTRIGGERREQ (register 0x3BA, offset 0xEE8)
- Integration Register ITTRIGGERACK (register 0x3B9, offset 0xEE4)

Implications

When this erratum occurs, the PREADYDBG output from the CoreSight ETM-R4 is driven low until PRESETDBGn is asserted LOW. This means the APB bus is locked up and no further accesses can be made to the CoreSight ETM-R4 or any other peripherals on the same APB bus.

This erratum is highly unlikely to occur, since debug tools must always power up the ETM before accessing the affected registers. If NIDEN and DBGEN are both LOW, tools must inspect the Authentication Status register (register 0x3EE, offset 0xFB8) to determine if non-invasive debug is enabled before accessing the affected registers. The only other likely reason to access these registers is by faulty software, for example where a corrupted pointer causes the ETM registers to be accessed by accident.

Workaround

This is a workaround for tools vendors.

Tools should ensure that the proper mechanisms are used to detect if the ETM is powered up and non-invasive debug is enabled before accessing the affected registers. The ETMPWRDOWN bit (bit [0] of the ETM Control register, register 0x000) must be LOW and the Authentication Status register (register 0x3EE, offset 0xFB8) must indicate non-invasive debug is enabled.

There is no workaround when faulty software accidentally accesses the ETM.

400025: AFREADYM signal might not be asserted

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 2, Present in: r0p0, Fixed in r1p0.

Description

On the AMBA Trace Port (ATB) of the CoreSight ETM-R4, two signals are provided to implement a trace data flushing mechanism. AFVALIDM is an input signal which requests a flush of all trace currently stored in the CoreSight ETM-R4. AFREADYM is an output signal which indicates when all stored data has been output by the CoreSight ETM-R4 and is asserted in response to an assertion of AFVALIDM. This flushing mechanism allows trace capture devices to dynamically request all stored data to be output.

On CoreSight ETM-R4 the AFREADYM signal might not be asserted under certain conditions, causing the ETM to never acknowledge the flush request.

Two sets of conditions can cause this erratum to occur, described below in Conditions 1 and Conditions 2.

Conditions 1

The following conditions must occur in the following order for this erratum to occur:

1. The CoreSight ETM-R4 is reset using the nPORESET input
2. The ETMPWRDOWN bit (bit [0] of the ETM Control Register 0x000) is cleared
3. The ETMPWRDOWN bit is set
4. AFVALIDM is asserted

Conditions 2

The following conditions must occur in the following order for this erratum to occur:

1. The ETMPWRDOWN bit (bit [0] of the ETM Control Register 0x000) is cleared
2. The ETM is programmed and tracing is enabled
3. Both the NIDEN and DBGEN input signals are driven LOW while trace data is still in the ETM's FIFO
4. AFVALIDM is asserted

Implications

Trace capture devices will usually use the flush mechanism to request all data from the system before stopping trace capture. When this erratum occurs these trace capture devices will never receive a flush completion acknowledgement via AFREADYM and therefore might never stop capturing trace. For example, if the trace capture device is the CoreSight TPIU or CoreSight ETB then trace decompression tools might continue polling the TPIU or ETB waiting for trace capture to stop and it never stops, thereby causing an infinite loop until the device is reset, or a timeout in the tools is triggered.

Workaround

This is a workaround for tools vendors.

If a trace capture device is configured to issue a flush before trace capture stops then:

- If the CoreSight ETM-R4 is connected to a trace funnel, then the CoreSight ETM-R4 should be configured to be the lowest priority ATB source. This ensures that the ETM is flushed last. This might impact the number of FIFO overflows observed in a bandwidth limited system.
- The tools should monitor the status of the flush operation. When using a CoreSight TPIU or CoreSight ETB this can be done using the Formatter and Flush Status Register. If a flush is observed to continue for a long period of time (for example, 100ms or more) then trace capture should be disabled immediately without waiting for the flush to complete.

This might result in some of the trace which was present in the CoreSight ETM-R4 to not be captured by the trace capture device.

Trace tools can avoid this erratum by configuring the trace capture devices to never request a flush when the CoreSight ETM-R4 is connected to the trace capture device. This might result in some of the trace which was present in all trace sources to not be captured by the trace capture device.

450964: Trigger might not occur

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 2, Present in: r0p0, Fixed in r1p0.

Description

If an A-Sync packet is generated at the same time as a trigger occurs, the Trigger packet might not appear in the trace stream.

Additionally, the TRIGGER output signal from the ETM might not be asserted.

Conditions

The following conditions must occur in the same clock cycle:

1. An A-Sync packet is generated
2. A Trigger occurs

A-Sync packets are generated periodically in the trace stream and the period is dependent on the value of the Synchronization Frequency Register (register 0x078), which can take a value of between 72 and 4096.

Implications

The TRIGGER output signal might not be asserted. This means that the trigger condition might not occur on the trace port or be embedded in the trace stream by a Trace Port Interface Unit. This means trace capture and analysis tools might not stop capturing trace which means that the desired trace might be overwritten in the trace capture device. Additionally, any cross triggering in the system which is configured to trigger on the ETM's TRIGGER signal might not trigger.

It should be noted that this erratum is unlikely to occur. This is because A-Sync packets are usually generated quite rarely, depending on the Synchronization Frequency. For example, the default value of the Synchronization Frequency Register is 1024, which means that an A-Sync packet is generated every 1024 bytes of trace. Since the ETM can generate up to 4 bytes of trace per cycle, this would imply a 1 in 256 chance of the erratum occurring in a single trace run. However, since the trace output is bursty, with an average of around 2 bytes per cycle for full data trace, the erratum is likely to occur significantly less than once in 256 separate trace runs with the default Synchronization Frequency Register value (1024).

Workaround

The probability of this erratum occurring can be reduced by increasing the value of the Synchronization Frequency Register.

An External Output (EXTOUT) from the ETM could be used to trigger the cross-trigger infrastructure and the trace sink, such as a TPIU or ETB. It should be noted that the normal trigger generation in the ETM only generates one trigger event, whereas using the EXTOUT might cause multiple trigger events if used in this way. The ETM's sequencer could be used in conjunction with the EXTOUT to avoid this.

Errata - Category 3

379031: ETM address range comparator at bottom of memory does not match on unaligned wraparound access

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 3, Present in: r0p0, Fixed in r1p0.

Description

An address range comparator configured for data addresses must match if a data transfer accesses any byte within the range. If an unaligned transfer occurs at the top of memory which wraps around to the bottom of memory, the bytes accessed at the bottom of memory will be ignored by address range comparators.

This erratum only occurs when tracing code which performs Unpredictable operations. The ARM Architecture does not permit data accesses to wrap around the top of memory. These accesses are Unpredictable. However, the ETM is sometimes used to help find Unpredictable operations in software running on the ARM processor. It is therefore useful for the ETM to support this behavior.

This erratum only affects word and halfword transfers which wrap around the top of memory. A word-aligned multi-word transfer which wraps around the top of memory, for example and LDM or LDRD, is Unpredictable but is not affected by this erratum.

This erratum does not apply if the address range comparator has data value comparisons enabled. This is because unaligned accesses never match if data value comparisons are enabled.

Single address comparators are not affected.

Conditions

1. An address range comparator is configured for data addresses, with data value comparisons disabled.
2. A halfword access is performed at address 0xFFFFFFFF or a word access is performed at 0xFFFFFFFD or higher.
3. The access accesses bytes at the bottom of memory greater than or equal to the lower bound of the address range.
4. The access meets all other requirements for the address range comparator to match.

Example

Address range comparator 1 is configured as follows:

- match on loads or stores
- lower bound is 0x00000001
- upper bound is 0x00000002

TraceEnable is configured to start tracing whenever the address range comparator 1 matches.

A data transfer occurs of size word at address 0xFFFFFFFF. This accesses the following bytes:

- 0xFFFFFFFF
- 0x00000000
- 0x00000001
- 0x00000002

The access of the byte at address 0x00000001 should cause address range comparator 1 to match. Because of this erratum, it does not match, and the transfer is not traced.

Implications

The comparator does not match. If software is performing Unpredictable data transfers which wrap around the top of memory, accesses to addresses 0x00000000 to 0x00000002 might be missed.

Workaround

This workaround is intended for users of trace tools.

Where an address range comparator is selected to cause an event, configure the ETM such that the event also occurs on a match from the single address comparator which forms the lower bound of the address range.

For example:

- Single address comparator 1 is configured for data transfers at address 0x00000000.
- Single address comparator 2 is configured for data transfers at address 0x00000020.
- Address range comparator 1 is selected as an include region for TraceEnable, by setting bit 0 of register 0x09.

To work around this erratum, single address comparator 1 should also be selected as an include region for TraceEnable, by setting bit 0 of register 0x07.

Workaround implications

It might not always be possible to consider the single address comparator in the event programming. For example, if the trigger condition is (Address Range 1 AND Sequencer state 1) then it is not possible to work around this erratum, because an event can only consider two conditions.

Address range comparators with the 'exact match' bit clear, hold their value between data transfers. Single address comparators do not hold their value between data transfers. For example, if an event is configured, using this workaround, for (Address Range 1 OR Single address 1), then the event output will not hold its value after a wraparound access as

described above, because it relies on the Single Address Comparator matching. This might cause unexpected results, depending on the event being controlled.

379386: Lock Access can be modified when PADDRDBG31 is HIGH

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 3, Present in: r0p0, Fixed in r1p0.

Description

When PADDRDBG31 is HIGH, the CoreSight ETM-R4 should consider all programming accesses to have been initiated by an external debugger. When PADDRDBG31 is LOW, the CoreSight ETM-R4 should consider all programming accesses to have been initiated by software running on the system.

The lock access mechanism consists of the Lock Access Register (register 0x3EC) and the Lock Status Register (register 0x3ED). The Lock Status Register should indicate that no lock access mechanism exists if programming accesses are initiated from an external debugger, i.e. when PADDRDBG31 is HIGH. Writes to the Lock Access register should be ignored by the ETM when PADDRDBG31 is HIGH.

Due to this erratum on CoreSight ETM-R4, when PADDRDBG31 is HIGH, the Lock Status Register correctly indicates that no lock access mechanism is present on accesses from an external debugger. However the Lock Access Register can be used to mistakenly allow accesses by software running on the system. Also, the Lock Access Register can be used to mistakenly prevent accesses by software running on the system.

Conditions

1. PADDRDBG31 is HIGH, indicating an access from an external debugger
2. Debugger software writes to the Lock Access Register without checking the Lock Status Register

Implications

It is possible for target resident software to be given access to the ETM registers when accesses should be prevented using the lock access mechanism. Conversely, it is possible for target resident software to be prevented from accessing the ETM registers when accesses should be allowed by the lock access mechanism.

If an external debugger writes the lock access key, 0xC5ACCE55, to the Lock Access Register, the software lock is unlocked and software running on the system can write to the ETM registers. Additionally, if an external debugger was to write any value apart from 0xC5ACCE55 to the Lock Access Register, the software lock is locked, preventing any software running on the system from writing to the ETM registers.

Since an external debugger should read the Lock Status Register to determine if the lock mechanism is present before writing to the Lock Access Register, it is expected that no external debugger will write to the Lock Access Register thereby never causing this erratum to have any adverse effects.

Workaround

This is a workaround for tools vendors.

Tools must follow the recommended behavior for operating the lock access mechanism. Tools must read the Lock Status register before writing to the Lock Access Register. If the Lock Status Register indicates that no lock access mechanism exists, then the debugger must not write to the Lock Access Register.

This is the normal recommended behavior for tools, therefore in most cases no workaround is necessary.

401329: Power Down Status Register (PDSR) reads as zero

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 3, Present in: r0p0, Fixed in r1p0.

Description

The ETM Power Down Status Register (register 0xC5, offset 0x314) is used to indicate the current status of the ETM.

This register should read as 0x00000001 in CoreSight ETM-R4 indicating the ETM is always powered up, but it incorrectly reads as 0x00000000.

Implications

Tools read this register to determine if the ETM is accessible and whether register state has been lost due to a power down. The ETM-R4 does not support multiple power domains and this register should always read as 0x00000001, indicating the ETM is powered up and the registers are accessible. The read value of 0x00000000 incorrectly indicates the ETM is powered down and no registers are accessible. Tools might attempt to poll this register waiting for the ETM to be powered up but this condition will never be satisfied.

Workaround

This is a workaround for tools vendors.

When reading the Power Down Status Register on CoreSight ETM-R4, if the read is successful and no error response is detected ignore the value read and assume it was 0x00000001.

402614: Cycle count is up to 7 cycles greater than it should be on debug exit

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 3, Present in: r0p0, Fixed in r1p0.

Description

When the processor is in debug state tracing is automatically disabled on the ETM and so an I-Sync packet is generated by the ETM before the first item traced after the processor exits from debug state.

This erratum, which is unlikely to occur, causes the cycle count in the I-Sync packet generated by the ETM after the processor exits debug state to be up to 7 cycles greater than it should be. This erratum can occur when tracing is enabled on the ETM before and after the processor enters debug state and the ETM is configured in cycle accurate mode and the ETM FIFO overflows just before debug mode is entered.

Conditions

The specific conditions that cause this erratum to occur are as follows:

1. The ETM must be configured in cycle accurate mode and instruction tracing must be enabled (bit [12] set and bit [20] cleared of the ETM Control Register, 0x00).
2. The ETM is programmed and tracing is enabled.
3. There must be several consecutive cycles in which the processor executes an instruction that pass its condition codes. These cause the ETM to generate a format 1 P-header with a WE value greater than 1.
4. Then one cycle in which no instruction is executed. The ETM generates a format 3 P-header for this cycle. Then the processor enters debug state and a debug exception is signalled to the ETM.
5. The ETM's FIFO overflows due to the format 3 P-header generated in step 4.

Implications

If this erratum occurs the processor will appear to be in debug state for up to 7 cycles longer than it is in reality. All other trace information is unaffected.

Workaround

There is no workaround. This erratum is rare and minor, and so is expected not to be an issue for any users.

If the ETM is reprogrammed whilst the processor is halted then this erratum does not occur because the cycle count is cleared before trace is restarted.

445913: Address Range comparator is not cleared by a coprocessor access

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 3, Present in: r0p0, Fixed in r1p0.

Description

Under normal conditions, Address Range comparators configured for data address comparisons hold their state between accesses. If a coprocessor access occurs, the comparator should stop firing because there is no address to compare against. This erratum causes the ETM to ignore the coprocessor access and the address range comparator will continue to fire.

Conditions

The following sequence must occur in order:

1. A data address range comparator is configured with the exact match bit clear
2. A data transfer occurs inside the range, causing the comparator to fire
3. A coprocessor access occurs

The coprocessor access should clear the Address Range comparator, however this erratum causes the Address Range comparator to continue firing.

Implications

Any ETM resource which is configured to be sensitive to the comparator might behave unexpectedly, in that a counter might count for longer than expected or tracing might be disabled or enabled for longer than expected.

Trace is not corrupted by this erratum.

Workaround

There is no workaround for this erratum.

445931: PCLKENDBG not used in claim tag register

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 3, Present in: r0p0, Fixed in r1p0.

Description

The clock enable PCLKENDBG for the APB interface is not used on writes to the Claim Tag registers. This causes multiple writes to the Claim Tag registers.

Conditions

1. PCLKENDBG is used to slow down PCLKDBG
2. A write to the Claim Tag Set or Claim Tag Clear register is performed

Implications

This does not cause any functional problems with the operation of the ETM. The write data bus PWDATADBG must be stable for the whole duration of the APB transaction, which means that the same value is written multiple times.

This erratum might cause static timing analysis failures during implementation.

Workaround

This is a workaround for system implementors.

If a slow APB clock is required, PCLKDBG must be gated externally to the ETM to provide the required clock frequency. PCLKENDBG must be tied HIGH.

450966: Multiple trigger requests might occur

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 3, Present in: r0p0, Fixed in r1p0.

Description

The TRIGGER output signal from the ETM might be asserted multiple times for one trigger event.

Conditions

The following conditions must occur:

1. The TRIGSBYPASS input signal is tied LOW
2. A Trigger occurs
3. The ATB transaction which contains the Trigger packet is stalled by the CoreSight system for at least 5 cycles

The ATB transaction might be stalled because the trace port cannot output the data quickly enough. For example, if a trace port size of 4 bits is used, the ATB transaction might be delayed for up to 8 cycles. Other trace sources in the system might also affect the available bandwidth on the ATB interface.

Implications

The TRIGGER output signal from the ETM is asserted multiple times.

This might cause the CoreSight trace sink to indicate multiple triggers in the trace stream or on the trace port.

If the TRIGGER output signal is used in a cross trigger system then multiple trigger events might be signaled to other devices which are connected to the cross trigger system and configured to receive the trigger indication.

This does not affect the ETM's trace stream.

In most trace capture devices, this erratum should not be a problem.

Workaround

This is a workaround for tools vendors.

If the ETM trigger is being used to embed triggers into a formatted trace stream using the CoreSight ETB or CoreSight TPIU then multiple triggers might be embedded in the formatted trace stream. Only the first of these should be used and the others can be safely discarded.

If the ETM trigger is being used to indicate the trigger condition on a trace port driven by a CoreSight TPIU then multiple trigger conditions might be seen on the trace port. Only the first of these should be used and the others can be safely discarded.

Due to the low impact, there is no need to workaround this erratum.

550515: Consecutive flushes might not be acknowledged

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 3, Present in: r0p0,r1p0, Fixed in r2p0.

Description

Consecutive flushes occur when two flushes are performed on the ATB interface, one after the other, with no cycles between them.

The flush request and acknowledge signals on the ATB interface are AFVALID and AFREADY respectively. The AMBA 3 ATB Protocol Specification (ARM IHI 0032A) states that, when AFREADY is asserted, then AFVALID must be deasserted on the following cycle, unless an additional flush is intended.

The consequence of this erratum is that the second flush, of a consecutive flush sequence, will not complete if it is requested when TraceEnable is HIGH.

This erratum will not occur under any of the following conditions:

- The ProgBit is HIGH
- TraceEnable is LOW and the FIFO is empty
- The power down bit is HIGH
- The NIDEN and DBGEN inputs are LOW
- The Cortex-R4 is in the wait for interrupt state

Conditions

All of the following must occur:

1. A CoreSight system does not contain a trace funnel
2. The trace sinks are configured such that multiple flushes can occur in a trace session
3. A flush request is generated when another flush request is still outstanding

This erratum is unlikely to occur because normally only one flush request is outstanding at a time.

Implications

Trace sinks will usually use the flush mechanism to drain all data from the system before stopping trace capture. When this erratum occurs these trace sinks will never receive a flush completion acknowledgement and therefore might never stop capturing trace. For example, if the trace sink is either the CoreSight TPIU or CoreSight ETB, then trace debug tools might continue polling the TPIU or ETB waiting for trace capture to stop, causing an infinite loop until one of the conditions in the description that prevents the erratum occurs.

Workaround

This is a workaround for trace tool vendors. Normally a workaround is not required, but if it is, ensure that only one flush is generated per trace session.

728501: ETM-R4 fails to trace a VnT packet for the second half of a SWP instruction

Status

Affects: product CoreSight ETM-R4.

Fault status: Cat 3, Present in: r0p0,r1p0,r2p0, Fixed in r2p1.

Description

When tracing SWP or SWPB instructions, the load and store parts of the SWP or SWPB instruction should be traced with separate data packets to the same address. If the load transfer is traced and the store transfer is not traced, the store transfer should be traced with a “Value Not Traced” packet.

When this erratum occurs, erroneously, no trace is generated for the store transfer of the SWP or SWPB.

Conditions

The following conditions must occur:

1. The ETM is enabled and is tracing.
2. A SWP or SWPB instruction is executed.
3. ViewData is configured to only trace the load part of the SWP or SWPB instruction.

Implications

If the ETM traces any data transfer, a data packet must be traced for every subsequent data transfer for that same instruction. This allows trace analysis tools to determine which registers were used or updated by the traced data items. When this erratum occurs, only the load part of the SWP or SWPB instruction is traced and therefore analysis tools cannot determine if the transfer is the load or store part of the SWP or SWPB instruction. This might cause misinterpretation of the execution of the processor by the analysis tool.

The trace stream is not corrupted.

Workaround

These workarounds are for users or tool vendors.

- Ensure that for all SWP and SWPB instructions in your code ViewData is not configured to trace load data only
- If ViewData has been configured to trace only the load transfer of a SWP or SWPB instruction and a single transfer has been traced, the trace analyser can assume that this corresponds to the load part of the instruction