arm CHINA

Star SE (CN001) and Star SE with FPU (CN002)
Star SP (CN003) and Star SP with FPU (CN004)
Star CU (CN005) and Star CU with FPU (CN006)

Software Developer Errata Notice

This document contains all known errata since the r0p0 release of the product.

Non-Confidential Proprietary Notice

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 China and the party that Arm China delivered this document to.

Web address

http://www.armchina.com

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 this document

If you have comments on content, send an e-mail to Support-CPU@armchina.com giving:

- The document title.
- The document number: 00911001_0200_00_en.
- If applicable, the page number(s) to which your comments refer.
- A concise explanation of your comments.

Arm China also welcomes general suggestions for additions and improvements.

Contents

Introduction

Scope

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

- The current status of the erratum.
- Where the implementation deviates from the specification and the conditions required for erroneous behavior to occur.
- The implications of the erratum with respect to typical applications.
- The application and limitations of a workaround where possible.

Categorization of errata

Programmer Those errata that impact the end-user/programmer of the hardware product. Software might observe the error and might require a software workaround or avoidance.

Within the above class, errata are split into three levels of severity and further qualified as common or rare:

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.

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 erratum text. Fixed errata are not shown as updated unless the erratum text has changed. The errata summary table on page 7 identifies errata that have been fixed in each product revision.

26-Aug-2021: Changes in document version 2.0				
ID	Status	Area	Cat	Summary of erratum
0063875	New	Programmer	CatB	A partially completed VLLDM might leave secure floating-point data unprotected

11-Sep-2020: Changes in document version 1.0				
ID	Status	Area	Cat	Summary of erratum
0063600	New	Programmer	CatB	Store-Release to external PPB interface might complete before previous Store-Release
0062251	New	Programmer	CatC	DAP SW-DP DPIDR value is incorrect
0062584	New	Programmer	CatC	Update to NVIC-related register might cause incorrect NVIC behavior for one cycle

Errata summary table

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

ID	Area	Cat	Summary	Found in versions	Fixed in version
0063875	Programmer	CatB	A partially completed VLLDM might leave secure floating-point data unprotected	r0p1, r1p0	Open
0063600	Programmer	CatB	Store-Release to external PPB interface might complete before previous Store-Release	r0p1	r1p0
0062251	Programmer	CatC	DAP SW-DP DPIDR value is incorrect	r0p0	r0p1
0062584	Programmer	CatC	Update to NVIC-related register might cause incorrect NVIC behavior for one cycle	r0p0, r0p1	r1p0

Errata descriptions

Category A

There are no errata in this category.

Category A (rare)

There are no errata in this category.

Category B

0063600

Store-Release to external PPB interface might complete before previous Store-Release

Status

Affects: Star SP, Star SP with FPU, Star SE, Star SE with FPU
Fault Type: Programmer Category B
Fault Status: Present in r0p1. Fixed in r1p0.

Description

The Armv8-M architecture defines rules on the ordering between a Store-Release instruction and loads and stores that appear in program order before the Store-Release instruction.

Because of this erratum, a Store-Release instruction to the external PPB interface might complete before another Store-Release instruction that appears immediately before it in program order.

Configurations Affected

This erratum affects configurations of the Star processor that includes a cache or a TCM.

Conditions

This erratum occurs when the processor executes the following sequence of instructions back-to-back:

- Store-Release1 to shareable normal memory on the C-AHB or S-AHB interface.
- Store-Release2 to the external PPB interface.

In addition to the instruction sequence above, there are conditions on what happens before Store-Release1 is executed, so that Store-Release1 goes into the store buffer and does not drain out of the store buffer immediately.

Implications

Store-Release2 completes on the external PPB interface before Store-Release1 completes on the C-AHB or S-AHB interface, potentially causing issues if the program relies on the ordering between Store-Release1 and Store-Release2.

Workaround

Note that the sequence of instructions that triggers this erratum is not expected to be found in most applications.

To avoid triggering this erratum, you can insert a NOP between Store-Release1 and Store-Release2.

0063875

A partially completed VLLDM might leave secure floating-point data unprotected

Status

Affects: Star SE with FPU
Fault Type: Programmer Category B
Fault Status: Present in r0p1, r1p0. Open.

Description

The VLLDM instruction allows secure software to restore a floating-point context from memory. Due to this erratum, if this instruction is interrupted or faults (that is, it does not complete), the secure data might be left unprotected in the floating-point register file, including the FPSCR.

Configurations Affected

This erratum affects configurations of the Star processor configured with the Security Extension and the Floating-point Extension

Conditions

This erratum occurs if:

- There is no active floating-point context (CONTROL.FPCA==0).
- Secure lazy floating-point state preservation is not active (FPCCR_S.LSPACT==0).
- The floating-point registers are treated as secure (FPCCR_S.TS==1).
- Secure floating-point state needs to be restored (CONTROL_S.SFPA==1).
- Non-secure state is permitted to access the floating-point registers (NSACR.CP10==1).
- A VLLDM instruction has loaded at least one register from memory and does not complete due to an interrupt or fault.

Implications

If the floating-point registers contain secure data, a VLSTM instruction is usually executed before calling a non-secure function to protect the secure data. This might cause the data to be transferred to memory (either directly by the VLSTM, or indirectly by the triggering of a subsequent lazy state preservation operation). If the data has been transferred to memory, it is restored using VLLDM on return to Secure state.

If VLLDM is interrupted or faults before it completes and enters a Non-secure handler, the partial register state which has been loaded will be accessible to Non-secure state.

Workaround

Software can ensure that a floating-point context is active before executing the VLLDM instruction by performing the following sequence with FPCCR_S.ASPEN set to 1:

- Read CONTROL_S.SFPA.
- If CONTROL_S.SFPA==1, then execute an instruction which has no functional effect apart from causing context creation (for example, VMOV S0, S0).

Category B (rare)

There are no errata in this category.

Category C

0062251

DAP SW-DP DPIDR value is incorrect

Status

Affects: Star CU, Star CU with FPU
Fault Type: Programmer Category C
Fault Status: Present in r0p0. Fixed in r0p1.

Description

The Star *Debug Access Port* (DAP) enables a debugger to connect to the Star processor using one of two optionally supported protocols: JTAG and *Serial Wire* (SW). One step in the connection process is for the debugger to read the *Debug Port* (DP) *ID Register* (DPIDR).

Because of this erratum, the value of the DESIGNER field in the SW-DP DPIDR is incorrect. This might cause a debugger using the SW protocol to misidentify the Star DAP or to fail to connect to the Star DAP.

Configurations Affected

This erratum affects configurations of the Star DAP that includes support for the SW protocol.

Conditions

This erratum occurs when the following conditions are met:

- The debugger tries to connect using the SW protocol.
- The debugger only recognizes several specific values for the DPIDR.

Implications

A debugger might misidentify the Star DAP or fail to connect to the Star DAP.

Workaround

You can add support in the debugger to recognize the incorrect value.

If your implementation supports both JTAG and SW, then you can connect to the Star DAP using the JTAG protocol.

0062584

Update to NVIC-related register might cause incorrect NVIC behavior for one cycle

Status

Affects: Star CU, Star CU with FPU, Star SP, Star SP with FPU, Star SE, Star SE with FPU
Fault Type: Programmer Category C
Fault Status: Present in r0p0, r0p1. Fixed in r1p0.

Description

Several registers in the NVIC are used to determine execution priority, interrupt priority, and in turn exception preemption, fault escalation and other NVIC behaviors: ITNS, IPR, ICER, ISER and AIRCR.

When some interrupts are configured as higher-latency and such a higher-latency interrupt is pending and one of the NVIC registers previously listed is changed, there is a one-cycle window where execution and interrupt priority might be incorrect, leading to incorrect NVIC behaviors such as exception preemption and fault escalation.

Configuration Affected

This erratum affects configurations where some interrupts are configured as higher-latency.

Conditions

This erratum occurs when the following conditions are met:

- A higher-latency interrupt is active or pending.
- The AIRCR or this higher-latency interrupt's target (ITNS), raw priority (IPR) or enable (ICER and ISER) register is changed.

There are additional conditions on the prioritization status of the higher-latency interrupt before and after the NVIC register write.

Implications

This erratum can lead to the following incorrect NVIC behaviors:

- The handler of a pending and enabled interrupt might be entered erroneously.
- Synchronous fault escalation to HardFault might be incorrect when a synchronous fault is raised in the cycle after the write to the NVIC-related register.
- The *RDY bits (that is, UFRDY) in FPCCR might be incorrect when the write to the NVIC-related register is followed immediately by a VLSTM instruction.
- Instruction stepping, asynchronous debug events, and breakpoints might be incorrectly triggered or missed in the cycle after the write to the NVIC-related register.

Workaround

There is no workaround for this erratum.

Note that typical applications do not need a workaround because registers related to interrupt priority are typically programmed during boot-up and then remain static.