This document contains all known errata since the r0p0 release of the product.
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 © 2019 Arm Limited (or its affiliates). All rights reserved.


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.

Web address

http://www.arm.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 then send an e-mail to errata@arm.com giving:

- The document title.
- The document number: SDEN-854652.
- 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.
Contents

INTRODUCTION

ERRATA SUMMARY TABLE

1190471 Core power down might cause data corruption
795148 DSU might fail to respond to a snoop
791818 DSU might snoop a core being powered off
773120 Combinations of external snoops and L1/L2 linefills can cause loss of coherency
1116019 WFE event might be missed in a multithreaded core
1147200 Disabling SIMD retention support while in retention might cause deadlock
1162044 Incorrect ordering of Clean to the Point of Persistence
850423 MV/PN bits in ERR1STATUS not implemented as write-one-to-clear
848504 Debug request trigger event may fail to halt a core leaving reset
798953 DSU clock gating might miss transfers on ACE master or Peripheral ports
814818 Allocating streaming write might deadlock
761074 Data corruption to a cacheable line in the lowest 256k of physical address space
936184 DSU might lose ACP transactions during clock gating
1249448 Poison information might get lost in CHI.C direct connect configurations
1219898 DSU might fail to detect ECC error
1314123 Incorrect ordering after change in cacheability
1299953 DSU might not report Uncontainable error on atomic instruction
874812 Cluster ELA/CTI ROM table entries present when ELADISABLE is high
824788 Error record overflow field increments on incorrect event
792397 Reading some ROM table registers always returns 0 when v7 memory map is used
774763 ECC errors in LTDB RAMs can cause data corruption and/or deadlock
776914 CHAIN PMU event counts incorrectly
766359 ERR1PFGCTRL might inject wrong fault type
787516 ECC errors in LTDB RAMs can cause spurious reports of correctable errors
1580900 Incorrect EDPFR value

SDEN-854652 Copyright © 2019, Arm Limited or its affiliates. All rights reserved. Page 4 of 36 Non-confidential
r3p0 implementation fixes

Note the following errata might be fixed in some implementations of r3p0. This can be determined by reading the CLUSTERREVIDR register where a set bit indicates that the erratum is fixed in this part.

| REVIDR[0] | 1190471 Core power down might cause data corruption |

Note that there is no change to the CLUSTERIDR which remains at r3p0 but the CLUSTERREVIDR is updated to indicate which errata are corrected. Software will identify this release through the combination of CLUSTERIDR/CLUSTERIDR_EL1 and CLUSTERREVIDR/CLUSTERREVIDR_EL1.

r2p0 implementation fixes

Note the following errata might be fixed in some implementations of r2p0. This can be determined by reading the CLUSTERREVIDR register where a set bit indicates that the erratum is fixed in this part.

| REVIDR[0] | 1190471 Core power down might cause data corruption |

Note that there is no change to the CLUSTERIDR which remains at r2p0 but the CLUSTERREVIDR is updated to indicate which errata are corrected. Software will identify this release through the combination of CLUSTERIDR/CLUSTERIDR_EL1 and CLUSTERREVIDR/CLUSTERREVIDR_EL1.

r0p1 implementation fixes

Note the following errata might be fixed in some implementations of r0p1. This can be determined by reading the CLUSTERREVIDR register where a set bit indicates that the erratum is fixed in this part.

| REVIDR[1] | 848504 Debug request trigger event may fail to halt a core leaving reset |

Note that there is no change to the CLUSTERIDR which remains at r0p1 but the CLUSTERREVIDR is updated to indicate which errata are corrected. Software will identify this release through the combination of CLUSTERIDR/CLUSTERIDR_EL1 and CLUSTERREVIDR/CLUSTERREVIDR_EL1.

r0p0 implementation fixes

Note the following errata might be fixed in some implementations of r0p0. This can be determined by reading the CLUSTERREVIDR register where a set bit indicates that the erratum is fixed in this part.

| REVIDR[0] | 773120 Combinations of external snoops and L1/L2 linefills can cause loss of coherency |

Note that there is no change to the CLUSTERIDR which remains at r0p0 but the CLUSTERREVIDR is updated to indicate which errata are corrected. Software will identify this release through the combination of CLUSTERIDR/CLUSTERIDR_EL1 and CLUSTERREVIDR/CLUSTERREVIDR_EL1.
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

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 10 identifies errata that have been fixed in each product revision.

### 23-Oct-2019: Changes in document version 17.0

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1580900</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>Incorrect EDPFR value</td>
</tr>
</tbody>
</table>

### 12-Aug-2019: Changes in document version 16.0

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1314123</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>Incorrect ordering after change in cacheability</td>
</tr>
</tbody>
</table>

### 19-Nov-2018: Changes in document version 15.0

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1299953</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>DSU might not report Uncontainable error on atomic instruction</td>
</tr>
</tbody>
</table>

### 19-Sep-2018: Changes in document version 14.0

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1162044</td>
<td>New</td>
<td>Programmer</td>
<td>CatB</td>
<td>Incorrect ordering of Clean to the Point of Persistence</td>
</tr>
<tr>
<td>1249448</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>Poison information might get lost in CHI.C direct connect configurations</td>
</tr>
<tr>
<td>1219898</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>DSU might fail to detect ECC error</td>
</tr>
</tbody>
</table>

### 20-Jul-2018: Changes in document version 13.0

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1190471</td>
<td>New</td>
<td>Programmer</td>
<td>CatA</td>
<td>Core power down might cause data corruption</td>
</tr>
<tr>
<td>1116019</td>
<td>New</td>
<td>Programmer</td>
<td>CatB</td>
<td>WFE event might be missed in a multithreaded core</td>
</tr>
<tr>
<td>1147200</td>
<td>New</td>
<td>Programmer</td>
<td>CatB</td>
<td>Disabling SIMD retention support while in retention might cause deadlock</td>
</tr>
</tbody>
</table>

### 22-Jan-2018: Changes in document version 12.0

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or updated errata in this document version.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 16-Oct-2017: Changes in document version 11.0

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or updated errata in this document version.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Status</td>
<td>Area</td>
<td>Cat</td>
<td>Summary of erratum</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>----------</td>
<td>---------</td>
<td>--------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No new or updated errata in this document version.</td>
</tr>
</tbody>
</table>

**07-Aug-2017: Changes in document version 10.0**

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>936184</td>
<td>New</td>
<td>Programmer</td>
<td>CatB</td>
<td>DSU might lose ACP transactions during clock gating</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>874812</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>Cluster ELA/CTI ROM table entries present when ELADISABLE is high</td>
</tr>
</tbody>
</table>

**26-May-2017: Changes in document version 8.0**

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>814818</td>
<td>Updated</td>
<td>Programmer</td>
<td>CatB</td>
<td>Allocating streaming write might deadlock</td>
</tr>
</tbody>
</table>

**12-May-2017: Changes in document version 7.0**

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>850423</td>
<td>New</td>
<td>Programmer</td>
<td>CatB</td>
<td>MV/PN bits in ERR1STATUS not implemented as write-one-to-clear</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>848504</td>
<td>New</td>
<td>Programmer</td>
<td>CatB</td>
<td>Debug request trigger event may fail to halt a core leaving reset</td>
</tr>
<tr>
<td>824788</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>Error record overflow field increments on incorrect event</td>
</tr>
</tbody>
</table>

**31-Jan-2017: Changes in document version 5.0**

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>795148</td>
<td>New</td>
<td>Programmer</td>
<td>CatA (rare)</td>
<td>DSU might fail to respond to a snoop</td>
</tr>
<tr>
<td>791818</td>
<td>New</td>
<td>Programmer</td>
<td>CatA (rare)</td>
<td>DSU might snoop a core being powered off</td>
</tr>
<tr>
<td>798953</td>
<td>New</td>
<td>Programmer</td>
<td>CatB</td>
<td>DSU clock gating might miss transfers on ACE master or Peripheral ports</td>
</tr>
<tr>
<td>814818</td>
<td>New</td>
<td>Programmer</td>
<td>CatB</td>
<td>Allocating streaming write might deadlock</td>
</tr>
<tr>
<td>ID</td>
<td>Status</td>
<td>Area</td>
<td>Cat</td>
<td>Summary of erratum</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>----------</td>
<td>------</td>
<td>------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>792397</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>Reading some ROM table registers always returns 0 when v7 memory map is used</td>
</tr>
</tbody>
</table>

**05-Dec-2016: Changes in document version 4.0**

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>774763</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>ECC errors in LTDB RAMs can cause data corruption and/or deadlock</td>
</tr>
<tr>
<td>776914</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>CHAIN PMU event counts incorrectly</td>
</tr>
<tr>
<td>766359</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>ERR1PGCTLR might inject wrong fault type</td>
</tr>
<tr>
<td>787516</td>
<td>New</td>
<td>Programmer</td>
<td>CatC</td>
<td>ECC errors in LTDB RAMs can cause spurious reports of correctable errors</td>
</tr>
</tbody>
</table>

**09-Nov-2016: Changes in document version 3.0**

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>773120</td>
<td>New</td>
<td>Programmer</td>
<td>CatA</td>
<td>Combinations of external snoops and L1/L2 linefills can cause loss of coherency</td>
</tr>
</tbody>
</table>

**13-Oct-2016: Changes in document version 2.0**

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td>761074</td>
<td>New</td>
<td>Programmer</td>
<td>CatB</td>
<td>Data corruption to a cacheable line in the lowest 256k of physical address space</td>
</tr>
</tbody>
</table>

**06-Oct-2016: Changes in document version 1.0**

<table>
<thead>
<tr>
<th>ID</th>
<th>Status</th>
<th>Area</th>
<th>Cat</th>
<th>Summary of erratum</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No errata in this document version.</td>
</tr>
</tbody>
</table>
## Errata summary table

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

<table>
<thead>
<tr>
<th>ID</th>
<th>Cat</th>
<th>Summary</th>
<th>Found in versions</th>
<th>Fixed in version</th>
</tr>
</thead>
<tbody>
<tr>
<td>1190471</td>
<td>CatA</td>
<td>Core power down might cause data corruption</td>
<td>r0p0, r0p1, r1p0, r0p2, r2p0, r3p0</td>
<td>r4p0</td>
</tr>
<tr>
<td>795148</td>
<td>CatA</td>
<td>DSU might fail to respond to a snoop</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>791818</td>
<td>CatA</td>
<td>DSU might snoop a core being powered off</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>773120</td>
<td>CatA</td>
<td>Combinations of external snoops and L1/L2 linefills can cause loss of coherency</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>1116019</td>
<td>CatB</td>
<td>WFE event might be missed in a multithreaded core</td>
<td>r3p0</td>
<td>r4p0</td>
</tr>
<tr>
<td>1147200</td>
<td>CatB</td>
<td>Disabling SIMD retention support while in retention might cause deadlock</td>
<td>r0p0, r0p1, r1p0, r0p2, r2p0, r3p0</td>
<td>r4p0</td>
</tr>
<tr>
<td>1162044</td>
<td>CatB</td>
<td>Incorrect ordering of Clean to the Point of Persistence</td>
<td>r0p0, r0p1, r1p0, r2p0, r3p0, r4p0</td>
<td>Open</td>
</tr>
<tr>
<td>850423</td>
<td>CatB</td>
<td>MV/PN bits in ERR1STATUS not implemented as write-one-to-clear</td>
<td>r0p0, r0p1</td>
<td>r1p0, r0p2</td>
</tr>
<tr>
<td>848504</td>
<td>CatB</td>
<td>Debug request trigger event may fail to halt a core leaving reset</td>
<td>r0p0, r0p1</td>
<td>r1p0, r0p2</td>
</tr>
<tr>
<td>798953</td>
<td>CatB</td>
<td>DSU clock gating might miss transfers on ACE master or Peripheral ports</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>814818</td>
<td>CatB</td>
<td>Allocating streaming write might deadlock</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>761074</td>
<td>CatB</td>
<td>Data corruption to a cacheable line in the lowest 256k of physical address space</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>936184</td>
<td>CatB</td>
<td>DSU might lose ACP transactions during clock gating</td>
<td>r0p1, r1p0, r0p2</td>
<td>r2p0</td>
</tr>
<tr>
<td>1249448</td>
<td>CatC</td>
<td>Poison information might get lost in CHI.C direct connect configurations</td>
<td>r2p0, r3p0</td>
<td>r4p0</td>
</tr>
<tr>
<td>1299953</td>
<td>CatC</td>
<td>DSU might lose ACP transactions during clock gating</td>
<td>r0p0, r0p1, r1p0, r0p2, r2p0, r3p0</td>
<td>r4p0</td>
</tr>
<tr>
<td>874812</td>
<td>CatC</td>
<td>Cluster ELA/CTI ROM table entries present when ELADISABLE is high</td>
<td>r0p0, r0p1</td>
<td>r0p2</td>
</tr>
<tr>
<td>824788</td>
<td>CatC</td>
<td>Error record overflow field increments on incorrect event</td>
<td>r0p0, r0p1</td>
<td>r1p0, r0p2</td>
</tr>
<tr>
<td>792397</td>
<td>CatC</td>
<td>Reading some ROM table registers always returns 0 when v7 memory map is used</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>774763</td>
<td>CatC</td>
<td>ECC errors in LTDB RAMs can cause data corruption and/or deadlock</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>776914</td>
<td>CatC</td>
<td>CHAIN PMU event counts incorrectly</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>766359</td>
<td>CatC</td>
<td>ERR1PFGCTLR might inject wrong fault type</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>ID</td>
<td>Cat</td>
<td>Summary</td>
<td>Found in versions</td>
<td>Fixed in version</td>
</tr>
<tr>
<td>--------</td>
<td>-----</td>
<td>-------------------------------------------------------------------------</td>
<td>-------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>787516</td>
<td>CatC</td>
<td>ECC errors in LTDB RAMs can cause spurious reports of correctable errors</td>
<td>r0p0</td>
<td>r0p1</td>
</tr>
<tr>
<td>1580900</td>
<td>CatC</td>
<td>Incorrect EDPFR value</td>
<td>r4p0</td>
<td>r4p1</td>
</tr>
</tbody>
</table>
Errata descriptions

Category A

1190471
Core power down might cause data corruption

Status

Affects: DSU
Fault Type: Programmer Category A
Fault Status: Present in r0p0, r0p1, r0p2, r1p0, r2p0, and r3p0. Fixed in r4p0.

Description

If a core silently evicts a cache line and is later powered off, then the line might remain present in the DSU snoop filter. If there is further activity to this cache line while the core is powered off, then that can later cause a loss of cache coherency to that line when the core is powered on.

Configurations Affected

This erratum affects all configurations of the DSU when the DSU BROADCASTOUTER input pin is 1, indicating it is connected to a coherent interconnect, and either:

- The interconnect has a snoop filter.
- There is another fully coherent master connected to the interconnect.

Note that the types of cores present in the cluster affect the severity of this erratum. If no Cortex-A75 cores are configured, then this reduces the severity to a Category C erratum.

Conditions

1. A core performs a read of a cache line.
2. The core evicts the line from its L1 and L2 cache without notifying the DSU (this is referred to as a silent eviction).
3. The core is powered down.
4. No other cores in the cluster have a copy of the cache line.
5. Another core in the cluster or another master accesses the cache line.
6. The core is powered up.
7. Another core in the cluster or another master accesses the cache line.

If these conditions are met, in combination with the configurations affected, then in some cases the DSU might cause loss of coherency for the cache line between the DSU and another master connected to the interconnect.

Implications

If the erratum occurs, then the loss of coherency can lead to data corruption and in some systems potentially deadlock. One of the conditions required is a silent eviction from a core. The conditions and implications therefore also depend on the type of cores present:

- For Cortex-A75, silent evictions are possible when heavy memory traffic causes internal buffers to fill up.
- For all other types of cores, a silent eviction is only possible when some types of uncorrectable ECC errors are detected. Therefore, the only implications for these cores are a small increase in detected uncorrected error (DUE) failure in time (FIT) rate.

Workaround

There is no workaround for this erratum.
Category A (rare)

795148
DSU might fail to respond to a snoop

Status

Affects: DSU
Fault Type: Programmer Category A Rare
Fault Status: Present in r0p0. Fixed in r0p1.

Description

In rare situations, it is possible for the DSU to fail to respond to a snoop from the interconnect because of a livelock in the DSU, leading to a system deadlock.

Configurations Affected

This erratum affects systems that have a coherent interconnect connected to the ACE master port.

Conditions

1. The interconnect connected to the DSU ACE master port(s) sends a snoop to the DSU. The interconnect requires the snoop to make progress before it will make progress on certain transactions from the DSU.
2. There are transactions inside the DSU that incorrectly happen to block the snoop from making progress. This is most likely if these transactions all share the same L3 index, that is the lower bits of the cache-line address are the same.

Implications

If the erratum occurs, the DSU will not make any progress on the snoop transaction and so the system will deadlock. Note that this is categorized as rare on the assumption that the revisions affected are only used for engineering samples, and not production.

Workaround

There is no workaround for this erratum.
791818

DSU might snoop a core being powered off

Status

Affects: DSU
Fault Type: Programmer Category A Rare
Fault Status: Present in r0p0. Fixed in r0p1.

Description

Under rare conditions, it is possible for the DSU to send a snoop to a core that is being powered off, leading to deadlock.

Configurations Affected

This erratum affects all configurations of the DSU.
Note that the types of cores present in the cluster affect the severity of this erratum. If only Cortex-A55 cores are configured, then this reduces the severity to a Category C erratum.

Conditions

1. A core in the DSU cluster silently evicts a cache-line from its L1 and L2 cache without notifying the DSU (this is referred to as a silent eviction).
2. Later, the power controller uses the core P-channel interface to request to power off that core.
3. During the power off sequence, the DSU sends a snoop to the core after the core does not expect any more snoop requests.

Implications

If the erratum occurs, the core or cluster can deadlock. Note that this erratum is categorized as rare and is not expected to impact engineering samples.
One of the conditions required is a silent eviction from a core. The conditions and implications therefore also depend on the type of cores present:

- For Cortex-A55, a silent eviction is only possible when some types of uncorrectable ECC errors are detected. Therefore the only implications for Cortex-A55 are a small increase in detected uncorrected error (DUE) failure in time (FIT) rate.
- For Cortex-A75, silent evictions are also possible when heavy memory traffic causes internal buffers to fill up.

Workaround

There is no workaround for this erratum.
773120

Combinations of external snoops and L1/L2 linefills can cause loss of coherency

Status

Affects: DSU
Fault Type: Programmer Category A (Rare)
Fault Status: Present in r0p0. Fixed in r0p1.

Description

Under rare conditions, it is possible for the DSU to cause a loss of cache coherency, leading to data corruption and potentially deadlock.

Configurations Affected

This erratum affects all configurations of the DSU when connected to a coherent interconnect.
Note that the types of cores present in the cluster affect the severity of this erratum. If only Cortex-A55 cores are configured, then this reduces the severity to a Category C erratum.

Conditions

1. A core starts a linefill for a cache line at an address A0.
2. A core starts a linefill for a cache line at an address A1.
3. Address A1 is different to A0, but some of the lower address bits are the same so that they both map to the same set in the DSU snoop filter.
4. The set in the snoop filter is full with other addresses, so that the access to A1 causes a capacity eviction of the entry containing address A0.
5. The core that previously requested a linefill for the cache line at address A0 evicts the line from its L1 and L2 cache without notifying the DSU (this is referred to as a silent eviction).
6. A core requests a new linefill for address A0.
7. There is an external interconnect snoop for the cache line at address A0 at a time so that the previous linefill from the core for the same cache line is still outstanding in the interconnect.

If these conditions are met under certain timing conditions, the DSU might cause loss of coherency for the cache line at address A0. This loss of coherency can lead to data corruption as well as potential deadlock in the L1 and L2 memory systems of the connected cores.

Implications

If the erratum occurs, the loss of coherency can lead to data corruption and/or deadlock.
One of the conditions required is a silent eviction from a core. The conditions and implications therefore also depend on the type of cores present:

- For Cortex-A55, a silent eviction is only possible when some types of uncorrectable ECC errors are detected. Therefore the only implications for Cortex-A55 are a small increase in detected uncorrected error (DUE) failure in time (FIT) rate.
- For Cortex-A75, silent evictions are also possible when heavy memory traffic causes internal buffers to fill up.

Workaround

There is no workaround for this erratum.

Category B

1116019
WFE event might be missed in a multithreaded core

Status

Affects: DSU
Fault Type: Programmer Category B
Fault Status: Present in r3p0. Fixed in r4p0.

Description

When the DSU is configured with a multithreaded core, the state of the two threads can be controlled with the operating mode in the core P-Channel. If the multithreaded core is in the On power mode, then a WFE wakeup event to a thread can be lost if the second thread is being activated or deactivated using the COREPSTATEx[5:4] bits at the same time.

Configurations affected

This erratum only affects configurations that include a multithreaded core.

Conditions

1. The core is in the On power mode.
2. Thread 0 is in WFE state.
3. Thread 1 is being activated or deactivated using COREPSTATEx[5:4] bits.
4. A WFE wakeup event arrives for Thread 0 at the same time Thread 1 is undergoing the power mode transition.
   Alternatively, Thread 0 can be in WFE while Thread 1 is being activated or deactivated.

Implications

If the above conditions are met, then under specific microarchitectural timing conditions, Thread 0 will miss the wakeup event and remain in WFE.

The following wakeup events will still wake the thread up:

1. Physical and virtual interrupts, depending on the software settings in the core execution pipeline.
2. External debug request, when halting is allowed.
3. Any of the following events for the thread in WFE, if they arrive after the thread activation/deactivation for the second thread is complete:
   - An event from SEV instruction from any thread in the system.
   - An event sent by the timer event stream for the thread.
   - An event caused by the clearing of the global monitor for the thread.

Workaround

Software should program the timer to generate an event stream for the PE.
1147200
Disabling SIMD retention support while in retention might cause deadlock

Status
Affects: DSU
Fault Type: Programmer Category B
Fault Status: Present in r0p0, r0p1, r0p2, r1p0, r2p0, and r3p0. Fixed in r4p0.

Description
LITTLE cores in a DSU cluster support putting the SIMD/FP logic into a retention state. If this SIMD/FP retention mode is disabled by software while the SIMD/FP logic is already in retention and Core retention mode is enabled, then it can lead to a deadlock.

Configurations Affected
This erratum only affects LITTLE cores that implement functional retention support for the SIMD/FP logic.

Conditions
1. The CPUPWRCTRLR_EL1 SIMD_RET_CTLR field is programmed to a nonzero value.
2. The CPUPWRCTRLR_EL1 WFE_RET_CTLR or WFI_RET_CTLR fields are programmed to a nonzero value.
3. No SIMD or FP instructions are executed for a time longer than programmed in the CPUPWRCTRLR_EL1 register, so the power controller puts the core into the FUNC_RET power mode.
4. The CPUPWRCTRLR_EL1 SIMD_RET_CTLR field is programmed to zero.
5. A WFE or WFI instruction is executed.
6. The power controller puts the core into the FULL_RET power mode.
7. Some activity occurs that requires the core to leave FULL_RET. This could be a wakeup event or interrupt for the WFE/WFI, or could be a temporary wakeup, for example, to process a snoop.

Implications
The COREPACTIVEx[8] bit is set indicating the ON power mode is requested, however if the power controller requests to go directly from the FULL_RET mode to ON, then the DSU denies the request. This might lead to a deadlock.

Workaround
In many implementations, the SIMD retention enable is a static setting so no workaround is needed. If software needs to dynamically disable SIMD retention, then it should execute a SIMD or FP instruction immediately after writing to the CPUPWRCTRLR_EL1 SIMD_RET_CTLR field. This ensures that the SIMD/FP logic is not in retention by the time a WFI or WFE is executed. Note that interrupts might need to be disabled during this sequence to ensure that a WFI/WFE cannot be executed by a different context before the SIMD logic has left the retention state.
If a software workaround is not possible, then a hardware fix is possible. If the power controller makes a request to go directly from FULL_RET to ON, and the request gets unexpectedly denied, then it should instead try to go from FULL_RET to FUNC_RET, and then to ON, and the requests are accepted.
1162044
Incorrect ordering of Clean to the Point of Persistence

Status

Affects: DSU
Fault Type: Programmer Category B
Fault Status: Present in r0p0, r0p1, r0p2, r1p0, r2p0, r3p0, and r4p0. Open.

Description

When using Cortex-A55 cores, Cortex-A75 cores or multithreaded cores with certain uncommon memory types that are not cached in the cluster, the DSU might perform a data cache clean to the Point of Persistence before a store instruction to the same address, even when the Armv8.2 architecture requires that these instructions occur in program order. Other types of maintenance instruction are not affected.

Configurations Affected

This erratum affects configurations where:

- The DSU cluster contains at least one Cortex-A55 core, Cortex-A75 core or multithreaded core.
- The DSU input signal BROADCASTPERSIST=1, indicating that the system implements the Point of Persistence.

Conditions

1. A Cortex-A55 core, Cortex-A75 core or a multithreaded core executes a store instruction to memory that is one of the following memory types:
   - Inner Write-Back, Outer Write-Through.
   - Inner Write-Back, Outer Non-cacheable.
   - Inner Write-Through, Outer Write-Back.
   - Inner Write-Through, Outer Write-Through.
   - Inner Write-Through, Outer Non-cacheable.
2. The same core subsequently executes the AArch64 data cache clean to the Point of Persistence instruction (DC CVAP) to the same cache line address without a DMB or DSB instruction in between.

Implications

These memory types are not expected to be common, so most software should not be affected. If these memory types are used, then the Clean to the Point of Persistence might occur before the store instruction, so the cache maintenance operation does not guarantee that the store has reached the Point of Persistence. As a result, some stores might not have reached persistent memory when software believes they have, which might lead to corruption during powerdown or power failure.

The DSU master ports indicate that all these memory types are Non-cacheable, so there is no impact on any system cache.

Workaround

If possible, software should map persistent memory using the Inner Write-Back, Outer Write-Back or Inner Non-Cacheable, Outer Non-Cacheable memory types. If this is not possible, then software can work around this issue by inserting a DMB instruction before a DC CVAP instruction, which ensures that any previous stores are ordered before it.
850423

MV/PN bits in ERR1STATUS not implemented as write-one-to-clear

Status

Affects: DSU
Fault Type: Programmer Category B
Fault Status: Present in r0p0, r0p1. Fixed in r1p0 and r0p2.

Description

The DSU implements the ERR1STATUS system register to provide status information for the DSU error record. The MV and PN bits in that register are meant to be write-one-to-clear, but are incorrectly implemented as write-zero-to-clear.

Configurations Affected

All configurations of the DSU are affected.

Conditions

1. The ERR1STATUS MV bit or PN bit is 1.
2. Software writes to the ERR1STATUS. Note: the MV and PN bits ignore writes if any of ERR1STATUS. {CE, DE, UE} are set to 1, and the highest priority of these is not being cleared to 0 in the same write.
   - If the write value of the MV or PN bit is 0 and the bit was previously 1, the bit will be cleared, which is incorrect.
   - If the write value of the MV or PN bit is 1 and the bit was previously 1, the bit will not be cleared, which is incorrect.

Implications

The DSU will not clear the MV and PN bits when software expects. The DSU might clear the MV and PN bits when software does not expect.
This means that the PN bit might remain incorrectly set for later correctable or deferred errors, and software might infer that more Uncorrectable or Deferred errors are caused by poison than is the case.

Workaround

Software should treat the MV and PN bits as write-zero-to-clear instead of write-one-to-clear.
848504

Debug request trigger event might fail to halt a core leaving reset

Status

Affects: DSU
Fault Type: Programmer Category B
Fault Status: Present in r0p0 and r0p1. Fixed in r1p0 and r0p2.

Description

The debug request trigger event is an output trigger event from the CTI. It is asserted by the CTI to force a core into Debug state. If the debug request trigger event is asserted when the core is powered-off or in a Cold reset, then when the core leaves reset the trigger event might halt the wrong core.

Configurations Affected

The erratum affects all configurations of the DSU with more than one core present.

Conditions

1. A debug request trigger event is asserted for any core except core 0.
2. The core is powered-off or in a Cold reset.
3. The core leaves reset.

Implications

The trigger will be sent to core 0 rather than the intended core, which results in the intended core not entering Debug state when it leaves reset. This impacts the ability to debug over powerdown scenarios.

Workaround

When the CTI trigger is programmed, the DBGPRCR_EL1.CORENPDRQ or EDPRCR.CORENPDRQ bit should be set on each core. This prevents the core from powering off, instead going to the emulated power off state. When the core leaves the emulated power off state, it will be warm reset rather than cold reset, and so will not trigger the erratum.

An alternative workaround is also available that avoids emulated power off. The debugger can set the EDECR.RCE bit on all cores to enable the Reset Catch debug event. This will cause all cores to enter Debug state every time they leave a Cold reset. The debugger can then read the CTITRIGOUTSTATUS[0] bit to determine whether the core should remain in Debug state because of a cross trigger, or it should exit from Debug state and continue to boot normally. This will have an impact on performance when a core boots up.
798953
DSU clock gating might miss transfers on ACE master or Peripheral ports

Status
Affects: DSU
Fault Type: Programmer Category B
Fault Status: Present in r0p0. Fixed in r0p1.

Description
Under certain near idle conditions, it is possible for the DSU to miss response transfers on the ACE master port or Peripheral port, leading to deadlock.

Configurations Affected
This erratum affects all configurations of the DSU.

Conditions
1. The DSU issues one or more transactions on the ACE master port or the Peripheral port.
2. The DSU is otherwise idle, and attempts to gate the clock internally.
3. The system returns a response on the B-channel or R-channel while the DSU is clock-gated and so the DSU does not notice the response.

Implications
If the erratum occurs, the DSU will not notice the response and so the system can deadlock.

Workaround
This erratum can be prevented by software writing the DSU register CLUSTERACTLR[15] to 0b1 to disable high-level clock gating of the DSU. This will increase the power consumption of the DSU when idle. Note that there is a small risk of encountering this erratum before the register write is performed.
814818
Allocating streaming write might deadlock

Status

Affects: DSU
Fault Type: Programmer Category B
Fault Status: Present in r0p0. Fixed in r0p1.

Description

In rare situations, it is possible for the DSU to fail to complete a streaming write transaction, leading to a system deadlock.

Configurations Affected

This erratum affects systems that have a coherent interconnect connected to the ACE master port.

Conditions

1. A core executes a full cache line of stores and these do not allocate into the L1 or L2 caches in the core, but do allocate into the L3 cache in the DSU.
2. Other microarchitectural timing conditions occur.
3. Another transaction is executed to the same L3 index, or a request is made to power down the cluster.

Implications

If the erratum occurs, the subsequent transaction or powerdown will not complete, and so the system will deadlock.

Workaround

For Cortex-A55 cores, you must ensure that for Normal Inner Write-Back Outer Write-Back memory the Outer Write-Allocate policy is No Allocate. This can be done by programming the Memory Attribute Indirection Registers appropriately. Note that if using virtualization then trapping accesses to these registers may impact the ability to run such workloads.
For Cortex-A75 cores, you must program the Cortex-A75 register field CPUECTLR.L3_STREAM to 0b11.
761074
Data corruption to a cacheable line in the lowest 256K of physical address space

Status
Affects: DSU
Fault Type: System Category B
Fault Status: Present in r0p0. Fixed in r0p1.

Description
If a combination of multiple linefills occurs to different but related addresses, then it can cause data corruption if one of the linefills is in the lowest 256K of physical address space.

Configurations Affected
This erratum affects all configurations of the DSU.

Conditions
1. One or more pages in the range 0x0 to 0x3FFFF of the physical address map are marked as writeback cacheable memory.
2. A core performs linefills to at least three different addresses that map to the same index in the DSU snoop filter and cause the snoop filter to perform at least two back invalidations because of reaching capacity at that index.
3. The interconnect sends a snoop to the same index as one of the linefills.
4. There is a dependency in the interconnect so that one of the linefills cannot complete until after the snoop has completed.

Implications
Some SoCs might have peripherals or other non-memory components at the lowest parts of the address map, and therefore will not be affected by this erratum. For those SoCs that are affected, this erratum can result in data corruption to a cacheable line in the lowest 256K of physical address space.

Workaround
If a workaround is required, then the firmware or OS must ensure that the lowest 256K of physical address space is never marked as writeback cacheable memory.
Note that if this address range contained flash memory containing boot code, then the memory could be marked as Write-Through cacheable. This would work around this erratum and still allow it to be cached in the instruction cache but not in the data cache.
936184
DSU might lose ACP transactions during clock gating

Status
Affects: DSU
Fault Type: Programmer Category B
Fault Status: Present in r0p1, r0p2, and r1p0. Fixed in r2p0.

Description
Under certain near idle conditions, it is possible for the DSU to miss an address transfer on the ACP interface, leading to deadlock. In addition, if multiple transactions are sent with the same ACP ID then data corruption might occur.

Configurations Affected
This erratum only affects configurations of the DSU that contain the ACP interface.

Conditions
1. The DSU is idle, which allows the hardware to gate the DSU clock.
2. Around the same time, the DSU receives an ACP address transfer for a new transaction.

Implications
If the erratum occurs, the DSU will lose the last address transfer and so the system might deadlock. If another address transfer is sent on the same channel with the same ID, then data corruption might occur.

Workaround
The erratum can be prevented by software writing the DSU register CLUSTERACTLR[16:15] to 0b11 to disable high-level clock gating of the DSU. This will increase the power consumption of the DSU when idle.

Category B (rare)
There are no errata in this category.

Category C

1249448
Poison information might get lost in CHI.C direct connect configurations

Status
Affects: DSU
Fault Type: Programmer Category C
Fault Status: Present in r2p0 and r3p0. Fixed in r4p0.

Description
When the DSU is configured as CHI.C direct connect, the TraceTag information and the poison information on some parts of cache lines with uncorrectable ECC errors get lost when the data enters or leaves the core.

Configurations affected
This erratum only affects direct connect configurations with a core that has a CHI.C interface.

Conditions
Scenario 1
The TraceTag field is set in a snoop from the interconnect, and the snoop returns data.

Scenario 2

1. An uncorrectable ECC error is detected in any data cache in the system, either in the core or any other core or system cache.
2. This error causes the doubleword quantity to be marked as poisoned.
3. The error is in the fourth or eighth doubleword of the cache line.
4. The cache line is brought into the core as a linefill, or evicted from the core.

Implications

The TraceTag information is not propagated correctly, leading to reduced debug capabilities.

The poison information for the fourth and eighth doublewords of the cache line is incorrectly cleared, which can result in the erroneous data later being used without an abort being indicated.

There is still substantial benefit being gained from the ECC logic. This erratum might cause a small increase in overall system failure rate.

The detection of the errors within the core is still reported in the error record registers.

Workaround

There is no workaround.
1219898
DSU might fail to detect ECC error

Status

Affects: DSU
Fault Type: Programmer Category C
Fault Status: Present in r0p0, r0p1, r0p2, r1p0, r2p0, and r3p0. Fixed in r4p0.

Description

If a core silently evicts a cache line and then a snoop request from the interconnect accesses that cache line, then the DSU might not detect an ECC error in the L3 tag RAM or snoop filter RAM.

Configurations Affected

This erratum only affects configurations where the DSU is connected to a coherent ACE interconnect and the DSU is configured with ECC by setting SCU_CACHE_PROTECTION to TRUE.

Conditions

1. A core performs a read of a cache line.
2. The core evicts the line from its L1 and L2 cache without notifying the DSU. This is referred to as a silent eviction.
3. The ACE interconnect sends a snoop transaction to the DSU for the same cache line.
4. The DSU reads the L3 tag RAM or snoop filter RAM for the snoop transaction.
5. The L3 tag RAM or snoop filter RAM contains an ECC error at the index accessed.
6. There is at least one other snoop transaction outstanding.

Implications

If these conditions are met, then in some cases the DSU might not detect or report the single-bit or double-bit ECC error. This might cause a loss of coherency or data corruption and can lead to deadlock.

There is still substantial benefit being gained from the ECC logic.
This erratum might cause a negligible increase in overall system failure rate.

One of the conditions required is a silent eviction from a core. The conditions and implications therefore also depend on the type of cores present:

- For Cortex-A75, silent evictions are possible when heavy memory traffic causes internal buffers to fill up.
- For all other types of cores, a silent eviction is only possible when some types of uncorrectable ECC errors are detected in the core. Therefore, the only implications for these cores are a negligible increase in Detected Uncorrected Error (DUE) Failure In Time (FIT) rate.

Workaround

No workaround is required.
1314123

Incorrect ordering after change in cacheability

Status

Affects: DSU
Fault Type: Programmer Category C
Fault Status: Present in r0p0, r0p1, r0p2, r1p0, r2p0, r3p0, and r4p0. Open.

Description

If the memory type of an address region is changed from Cacheable to Non-cacheable, and then back again, and rare microarchitectural conditions occur, then stale data might be observed in a cache.

Configurations Affected

This erratum affects all configurations except Direct Connect configurations.

Conditions

1. An address region of memory is marked in the translation tables as Write-Back Cacheable memory.
2. The hardware prefetcher starts a data prefetch to an address within this region. This must generate a StashOnce CHI transaction from the core to the DSU, and in some cases the DSU might pass the StashOnce on to the interconnect if the DSU is configured with a CHI master.
3. The translation tables are updated to change the memory type to Non-cacheable or Device memory. This would involve a break-before-make sequence.
4. A sequence of cache clean and invalidate instructions are executed to ensure that any Cacheable data in the memory region does not remain in the caches.
5. The StashOnce transaction and the clean and invalidate transaction to the same address get reordered within the DSU or externally if the StashOnce was sent to the interconnect. This means that the StashOnce transaction can cause the line to be allocated into the cache after the cache maintenance has completed.
6. A core or other master in the system writes to the region that is now marked Non-cacheable or Device.
7. The translation tables are changed a second time, to mark the memory as Write-Back Cacheable again.
8. A load instruction is executed. The load might observe the stale data that was prefetched into the cache, rather than the Non-cacheable data that was written.

Implications

The above sequence is very specific and would typically take a very long time to execute. It requires that the StashOnce transaction is started before the translation table modification, yet does not complete until after both the translation table modification and the cache maintenance. Additionally, the StashOnce and cache maintenance transactions must be reordered by the DSU or interconnect, and this is an unlikely event, especially if they are not started at a similar time. Therefore the combination of these conditions is going to be extremely rare. Furthermore, the change in memory type implies a change of use of the memory, and many such changes of use will not require preservation of the data between uses.

Workaround

No workaround is necessary.

Note that this erratum is caused by a deficiency in the CHI architecture and will be corrected in CHI Issue D onward.
1299953

DSU might not report Uncontainable error on atomic instruction

Status

Affects: DSU
Fault Type: Programmer Category C
Fault Status: Present in r0p0, r0p1, r0p2, r1p0, r2p0, and r3p0. Fixed in r4p0.

Description

If the DSU receives read data from the interconnect where some but not all beats of the data indicate an error then, for certain types of atomic instruction, the DSU might not report the error.

Configurations Affected

This erratum affects all configurations except Direct Connect configurations. However if the system interconnect supports poisoning then it is unlikely to meet the other conditions required.

Conditions

1. A core executes an atomic store, atomic swap, or atomic compare instruction to Inner Write-Back, Outer Write-Back cacheable memory.
2. Either:
   - The DSU is configured with an ACE master port.
   - The DSU is configured with a CHI master port and the BROADCASTATOMIC input pin is LOW or the CLUSTERECTLR system register bit [7] is 1.
3. The DSU sends a 64-byte ReadUnique or ReadNoSnp transaction to the interconnect to fetch the data for the atomic instruction.
4. The interconnect returns data where the data needed by the atomic does not contain an error, but some of the other data beats in the same transaction indicate an error. On CHI, this means some data flits indicate DERR or NDERR. On ACE, this means some data transfers indicate SLVERR or DECERR.

Implications

If these conditions are met, then the DSU will discard the data from the interconnect, so data in the same cache line might be corrupted, potentially causing the loss of data previously written to that cache line. The atomic instruction will also not update memory. However, the destination register of the atomic instruction will be updated with the correct data.

The DSU will not record the Uncontainable error in the error record register, ERR1STATUS, or signal an error recovery interrupt (nERRIRQ) or fault handling interrupt (nFAULTIRQ). Also, the core executing the atomic instruction will not receive an External abort or System Error interrupt. This means that software will continue executing, unaware that memory has been corrupted.

This erratum cannot occur if the system does not return errors for only some but not all beats of data for cacheable read transactions. It is expected that this will be the case for many systems. Systems using a CHI interface and configured with ECC support would be expected to poison data that got an uncorrectable error rather than return a DERR or NDERR on only some of the beats of the transaction. These systems using poison would not be impacted by this erratum.

There is still substantial benefit being gained from the ECC logic, if configured.
In systems that meet the configurations described, this erratum might cause a negligible increase in overall system failure rate.

Workaround

No workaround is necessary.
874812
Cluster ELA/CTI ROM table entries present when ELADISABLE is HIGH

Status
Affects: DSU
Fault Type: Programmer Category C
Fault Status: Present in r0p0, r0p1. Fixed in r0p2.

Description
The DSU supports an integrated ELA-500 component in the cluster. If the ELA is present but is disabled by the ELADISABLE pin, then reading the ROM table entries would incorrectly indicate that the ELA and associated cluster CTI are present.

Configurations Affected
This erratum only affects configurations of the DSU with the cluster ELA present.

Conditions
1. The ELADISABLE pin is HIGH.
2. A debug APB read is made to ROMENTRY0/ROMENTRY1 (when the LEGACY_V7_DEBUG_MAP configuration is FALSE) or ROMENTRY5/ROMENTRY6 (when the LEGACY_V7_DEBUG_MAP configuration is TRUE).

Implications
The ROM table will indicate that the ELA and associated CTI are present, but any attempted access to them will read as zero and ignore writes.

Workaround
Any discovery code must be written to be tolerant of these components not being found.
824788
Error record overflow field increments on incorrect event

Status

Affects: DSU
Fault Type: Programmer Category C
Fault Status: Present in r0p0, r0p1. Fixed in r1p0 and r0p2.

Description

The DSU contains a register field, ERR1STATUS.OF, that is set when the error record has overflowed with multiple errors. If the DSU detects multiple Deferred or Uncorrected errors, the DSU sets ERR1STATUS.OF to 1, which is correct. If the DSU detects multiple Corrected errors, the DSU should only increment the Corrected error count, ERR1MISC0.CECR or ERR1MISC0.CECO, but the DSU also sets the ERR1STATUS.OF to 1, which is incorrect. The DSU also fails to set the ERR1STATUS.OF when one of the Corrected error counters, ERR1MISC0.CECR or ERR1MISC0.CECO, overflows.

Configurations Affected

This erratum only affects configurations of the DSU with SCU_CACHE_PROTECTION enabled.

Conditions

The ERR1STATUS.OF becomes 1 incorrectly when:

1. The ERR1STATUS register indicates a valid entry (ERR1STATUS.V=0b1) and the relevant counter in ERR1MISC0 (ERR1MISC0.CECR or ERR1MISC0.CECO) is not at its maximum value (ERR1MISC0.CECR!=0xFF or ERR1MISC0.CECO!=0xFF).
2. A Corrected error is detected.

The DSU fails to set ERR1STATUS.OF to 1 either when:

1. The ERR1STATUS register is marked as invalid (ERR1STATUS.V=0b0) and the ERR1MISC0.CECR is at the maximum value (ERR1MISC0.CECR=0xFF).
2. A Corrected error is detected which matches ERR1MISC0.INDX and ERR1MISC0.WAY.

OR:

1. The ERR1STATUS register is marked as invalid (ERR1STATUS.V=0b0) and the ERR1MISC0.CECO is at the maximum value (ERR1MISC0.CECO=0xFF).
2. A Corrected error is detected which does not match ERR1MISC0.INDX and ERR1MISC0.WAY.

Implications

In the presence of multiple errors, the reporting of overflow can be inaccurate, which could lead to error handling software overestimating or underestimating the number of errors that have occurred. This could result in a negligible increase in the failure in time (FIT) rate.

Workaround

There is no workaround.
792397
Reading some ROM table registers always returns 0 when v7 memory map is used

Status
Affects: DSU
Fault Type: Programmer Category C
Fault Status: Present in r0p0. Fixed in r0p1.

Description
Reading some ROM table registers in the legacy v7 memory map will return incorrect values.

Configurations Affected
This erratum only affects configurations with the legacy v7 debug memory map configuration option LEGACY_V7_DEBUG_MAP set to TRUE.

Conditions
A read is made on the debug APB interface to one of the following ROM table registers: PRIDR0, DEVARCh, DEVID, DEVTYPE, DBGPCR, or DBGPSR.

Implications
Reading these registers will return 0, which might lead to a debugger misidentifying the components.

Workaround
There is no workaround for this erratum.
774763
ECC errors in LTDB RAMs can cause data corruption and/or deadlock

Status
Affects: DSU
Fault Type: Programmer Category C
Fault Status: Present in r0p0. Fixed in r0p1.

Description
Under rare conditions, single and double-bit errors detected by the DSU in the LTDB RAMs can lead to data corruption and potentially deadlock.

Configurations Affected
This erratum affects all configurations of the DSU with SCU_CACHE_PROTECTION enabled.

Conditions
1. A core starts an Instruction-side linefill or a non-cacheable read of more than 128 bits.
2. The DSU transfers data to the core in response to the above request using the LTDB RAMs.
3. An ECC error is detected on data read from the LTDB RAMs for the transfer.
4. The same core starts a second linefill or a non-cacheable read (of any length).
5. The second linefill does not hit in the L3 cache, and completes before the first linefill data has been accepted by the core.

If these conditions are met, in addition to certain rare timing conditions, the DSU might cause data corruption and/or deadlock.

Implications
There is still substantial benefit being gained from the ECC logic. There might be a negligible increase in overall system failure rate because of this erratum.
The LTDB RAMs are very small, and might typically be implemented with more robust bitcells than larger RAMs. Therefore, the probability of an ECC error on these RAMs is significantly less than on other RAMs in the design.

Workaround
No workaround is required for the majority of systems. For designs where RAS is of significant importance, this erratum can be worked around by setting bit [4] of CLUSTERACTLR_EL1. This will increase the latency of some transaction types and will have a small impact on performance.
**776914**

**CHAIN PMU event counts incorrectly**

**Status**

Affects: DSU  
Fault Type: Programmer Category C  
Fault Status: Present in r0p0. Fixed in r0p1.

**Description**

The DSU provides a CHAIN PMU event for odd-numbered counters, allowing two 32-bit counters to be paired to provide a 64-bit counter. Because of this erratum, the CHAIN event might count incorrectly.

**Configurations Affected**

All configurations are affected.

**Conditions**

1. One of the odd-numbered PMU event counters is configured to count the CHAIN event.  
2. The preceding even-numbered counter overflows.

If these conditions are met, the odd-numbered counter will increment every cycle until the overflow condition is cleared by writing to the CLUSTERPMOVSCCLR_EL1 register.

**Implications**

When using a pair of counters to give a 64-bit count, the upper 32 bits of the counter value will be incorrect.

**Workaround**

No workaround is available.
766359

ERR1PFGCTLR might inject wrong fault type

Status

Affects: DSU
Fault Type: Programmer Category C
Fault Status: Present in r0p0. Fixed in r0p1.

Description

The ERR1PFGCTLR register provides a mechanism for software to generate faults in the DSU. Software controls the type of fault generated by setting different bits in the register.

When software programs the ERR1PFGCTLR to generate a Deferred Error (DE), the DSU might generate a DE or generate an Uncontainable Error (UC).

When software programs the ERR1PFGCTLR to generate an Unrecoverable Error (UEU), the DSU will generate an Uncontainable Error (UC).

Configurations Affected

This erratum affects all configurations of the DSU.

Conditions

The Error Pseudo Fault Generation Control Register must be programmed to generate DE or UEU errors:

1. ERR1PFGCTRL[5] must be set to 0b1 or ERR1PFGCTRL[2] must be set to 0b1.
2. ERR1PFGCTRL[31] must be set to 0b1.

Implications

Software that is testing the behavior of DE or UEU errors might see an unexpected UC error. Software might not expect the error type reported in the Error Record Primary Status Register, ERR1STATUS. The Uncontainable Error might have more severe consequences to the software and system than a DE or UEU.

Workaround

Software must not enable UEU injection as the DSU never generates UEU errors. Therefore ERR1PFGCTRL[2] must be set to 0b0.

There is no workaround to prevent DE generation sometimes causing UC errors.
**787516**

**ECC errors in LTDB RAMs can cause spurious reports of correctable errors**

**Status**

Affects: DSU  
Fault Type: Programmer Category C  
Fault Status: Present in r0p0. Fixed in r0p1.

**Description**

Under rare conditions, single and double-bit errors detected by the DSU in the LTDB RAMs can lead to spurious reports of correctable errors by setting the ERR1STATUS_EL1.OF flag incorrectly.

**Configurations Affected**

This erratum affects all configurations of the DSU with SCU_CACHE_PROTECTION enabled.

**Conditions**

1. The DSU transfers read data to a core in response to a core read request, or write data to the external ACE interface as a result of a cache eviction or streaming write using the LTDB RAMs.
2. A single or double-bit ECC error is detected on data read from the LTDB RAMs for the transfer, or the transfer carried poison from an earlier ECC error on any RAM in the system.
3. At the same time, the DSU transfers read data to a core in response to a core read request.

If these conditions are met, in addition to certain rare timing conditions, the DSU might set the ERR1STATUS_EL1.OF overflow flag when only a single error was detected.

**Implications**

There is still substantial benefit being gained from the ECC logic. The erratum only causes overly pessimistic error reporting.

**Workaround**

No workaround is required for the majority of systems. For designs where RAS is of significant importance and overly pessimistic error reporting is undesirable, this erratum can be worked around by setting bit [4] of CLUSTERACTLR_EL1. This will increase the latency of some transaction types and will have a small impact on performance.
1580900

Incorrect EDPFR value

Status

Affects: DSU
Fault Type: Programmer Category C
Fault Status: Present in r4p0. Fixed in r4p1.

Description

The EDPFR register is an external debug register that provides information about implemented PE features. When the DSU is used with a core that implements the Armv8.4-A architecture, the EDPFR.GIC field will indicate that the GICv4 system register interface is supported even when the GICv4.1 system register interface is supported.

Configurations Affected

The DSU is configured with a core that supports the Armv8.4-A architecture and the GICDISABLE input pin is tied LOW.

Conditions

The external debugger reads the EDPFR register and uses the contents of the GIC field (bits [27:24]).

Implications

Arm does not expect the external debugger will need to use the contents of the GIC field. If it does, then it might incorrectly decide that GICv4.1 features are not supported by the core. The system register ID_AA64PFR0_EL1 is unaffected and provides the correct information about the GIC system register interface support implemented.

Workaround

No workaround is necessary.