

ARM GIC CoreLink™ GIC-500 Generic Interrupt Controller

Product Revision r1p1

Software Developers Errata Notice

Non-Confidential - Final

ARM®

Software Developers Errata Notice

Copyright © 2016 ARM. All rights reserved.

Non-Confidential Proprietary Notice

This document is protected by copyright and the practice or implementation of the information herein may be protected by one or more patents or pending 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.

This document is Non-Confidential but any disclosure by you is subject to you providing the recipient the conditions set out in this notice and procuring the acceptance by the recipient of the conditions set out in this notice.

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

Unless otherwise stated in the terms of the Agreement, this document is provided "as is". ARM makes no representations or warranties, either express or implied, included but not limited to, warranties of merchantability, fitness for a particular purpose, or non-infringement, that the content of this document is suitable for any particular purpose or that any practice or implementation of the contents of the document will not infringe any third party patents, copyrights, trade secrets, or other rights. Further, ARM makes no representation with respect to, and has undertaken no analysis to identify or understand the scope and content of such third party 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 LOSS, LOST REVENUE, LOST PROFITS OR DATA, SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF OR RELATED TO ANY FURNISHING, PRACTICING, MODIFYING OR ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Words and logos marked with ® or TM are registered trademarks or trademarks, respectively, of ARM Limited. Other brands and names mentioned herein may be the trademarks of their respective owners. Unless otherwise stated in the terms of the Agreement, you will not use or permit others to use any trademark of ARM Limited.

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.

In this document, where the term ARM is used to refer to the company it means "ARM or any of its subsidiaries as appropriate".

Copyright © 2016 ARM Limited 110 Fulbourn Road, Cambridge, England CB1 9NJ. All rights reserved.

Web Address

<http://www.arm.com>

Feedback on content

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

- the document title
- the document number, ARM-EPM-012079
- the page numbers to which your comments apply
- a concise explanation of your comments.

ARM also welcomes general suggestions for additions and improvements.

Release Information

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

4 May 2016: Changes in Document v3

Page	Status	ID	Cat	Rare	Summary of Erratum
12	New	855721	CatC		GICD_TYPER.CPUNumber!=0 when the ARE bits are fixed as RAO/WI

24 Sep 2015: Changes in Document v2

Page	Status	ID	Cat	Rare	Summary of Erratum
10	New	852676	CatC		Read of GITS_PIDR3 value may return incorrect value

3 Mar 2015: Changes in Document v1

Page	Status	ID	Cat	Rare	Summary of Erratum
7	New	838419	CatB		GICR_WAKER.Sleep might lose an LPI
8	New	838420	CatB		MOVALL might cause corruption or deadlock
9	New	838421	CatB		Access to 8192 bytes below base address specified in GICR_PROPBASER
10	New	838422	CatB		INVALL does not affect enabled LPIs

Contents

CHAPTER 1.	5
INTRODUCTION	5
1.1. Scope of this document	5
1.2. Categorization of errata	5
CHAPTER 2.	6
ERRATA DESCRIPTIONS	6
2.1. Product Revision Status	6
2.2. Revisions Affected	6
2.3. Category A	7
2.4. Category A (Rare)	7
2.5. Category B	7
838419: GICR_WAKER.Sleep might lose an LPI.....	7
838420: MOVALL might cause corruption or deadlock.....	8
838421: Access to 8192 bytes below base address specified in GICR_PROPBASER.....	9
838422: INVALID does not affect enabled LPIs.....	10
2.6. Category B (Rare)	10
2.7. Category C	10
852676: Read of GITS_PIDR3 value may return incorrect value.....	10
855721: GICD_TYPER.CPUNumber!=0 when the ARE bits are fixed as RAO/WI.....	12

Chapter 1.

Introduction

This chapter introduces the errata notice for the CoreLink™ GIC-500 Generic Interrupt Controller .

1.1. Scope of this document

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

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

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

1.2. Categorization of errata

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

Table 1 **Categorization of errata**

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

Chapter 2.

Errata Descriptions

2.1. Product Revision Status

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

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

2.2. Revisions Affected

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

This document includes errata that affect revisions upto r1p1 only.

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

Table 2 **Revisions Affected**

ID	Cat	Rare	Summary of Erratum	Revisions Affected		
				r0p0	r1p0	r1p1
855721	CatC		GICR_WAKER.Sleep might lose an LPI	X	X	X
852676	CatC		Read of GITS_PIDR3 value may return incorrect value		X	
838422	CatB		INVALL does not affect enabled LPIs	X		
838421	CatB		Access to 8192 bytes below base address specified in GICR_PROPBASER	X		
838420	CatB		MOVALL might cause corruption or deadlock	X		
838419	CatB		GICR_WAKER.Sleep might lose an LPI	X		

2.3. Category A

There are no errata in this category

2.4. Category A (Rare)

There are no errata in this category

2.5. Category B

838419: GICR_WAKER.Sleep might lose an LPI

Category B

Products Affected: GIC-500 Generic Interrupt Ctlr - TERM.

Present in: r0p0

Description

The GIC-500 can implement support for LPI interrupts. If software is using the LPI support, software can save the pending status of all pending LPIs to memory by setting the GICR_WAKER.Sleep bit to 1. When the GICR_WAKER.Quiescent bit becomes 1, the GIC-500 will guarantee that the pending tables in memory will contain all pending LPIs. This Sleep bit is a feature that the GIC-500 provides in addition to the features of the GICv3 architecture.

This erratum means that when Sleep is set to 1, the GIC-500 might lose an LPI when writing the pending LPIs to the pending tables.

Configurations Affected

This erratum affects all configurations where the ITS is present and LPIs are supported.

Conditions

The erratum might occur if software sets the GICR_WAKER.Sleep bit to 1.

Implications

If the erratum occurs, the pending tables will not contain a complete list of the LPIs that are still pending. This means that an LPI might be lost.

Workaround

There is no workaround for this erratum.

The Sleep bit is a feature that is not defined the GICv3 architecture. Therefore standard software will not use the Sleep bit.

838420: MOVALL might cause corruption or deadlock**Category B****Products Affected: GIC-500 Generic Interrupt Ctlr - TERM.****Present in: r0p0****Description**

The GIC-500 can implement an Interrupt Translation Service (ITS) that provides maintenance operations for LPI interrupts. One maintenance command that the ITS provides is MOVALL. This command can be used as part of an architecturally-defined sequence to move all the LPIs from one core to another core.

This erratum means that using a MOVALL command may cause a deadlock of the GIC-500, the corruption of LPIs or the loss of LPIs.

Configurations Affected

This erratum affects all configurations where the ITS is present and LPIs are supported.

Conditions

The erratum might occur if software writes a MOVALL command into the command queue of the ITS, and then the MOVALL command is executed by the ITS in the GIC-500.

Implications

If the erratum occurs, the system might deadlock either due to a hardware deadlock or due to software not getting an expected LPI. This erratum might also lead to data corruption if software receives an unexpected interrupt.

Workaround

You can ensure that the erratum does not occur by not using MOVALL commands. The MOVALL sequence is used to move all LPIs from one target to another. This functionality can be replicated by using a number of MOVI commands, which are not affected by this erratum. Instead of using the MOVALL sequence, you must instead issue one MOVI command for each LPI that is currently mapped to the target you want to move the LPIs from. This will move each LPI individually.

838421: Access to 8192 bytes below base address specified in GICR_PROPBASER**Category B****Products Affected: GIC-500 Generic Interrupt Ctlr - TERM.****Present in: r0p0****Description**

The GIC-500 can implement support for LPI interrupts. If software is using the LPI support, the GIC-500 will sometimes read from the LPI configuration table in main memory using the address provided in the GICR_PROPBASER.

This erratum means that the GIC-500 might read from the LPI configuration table at an offset of -8192 bytes, that is exactly 8192 bytes before the start of the LPI configuration table. It might also cause data corruption or deadlock in the connected processor, as well as deadlock of the GIC-500, the corruption of LPIs or the loss of LPIs.

Configurations Affected

This erratum affects all configurations where the ITS is present and LPIs are supported.

Conditions

The erratum might occur if software enables the LPI support by setting the EnableLPIs bit to 1 in one of the GICR_CTLR registers.

Implications

If the erratum occurs, the system might deadlock either due to a hardware deadlock or due to software not getting an expected LPI. This erratum might also lead to data corruption if software receives an unexpected interrupt.

Workaround

You can ensure that the erratum does not occur by allocating the memory 8192 bytes before the start of the LPI configuration table, and setting that memory location to be zero.

838422: INVALL does not affect enabled LPIs**Category B****Products Affected: GIC-500 Generic Interrupt Ctlr - TERM.****Present in: r0p0****Description**

The GIC-500 can implement an Interrupt Translation Service (ITS) that provides maintenance operations for LPI interrupts. One maintenance command that the ITS provides is INVALL. This command ensures that all LPIs on an ITS collection use the latest priority and enable bits from the LPI configuration table.

This erratum means that an INVALL command might not affect all LPIs that are enabled when the INVALL command executes. These LPIs are then not guaranteed to use the latest priority and enable settings.

Configurations Affected

This erratum affects all configurations where the ITS is present and LPIs are supported.

Conditions

The erratum might occur if software writes an INVALL command into the command queue of the ITS, and then the INVALL command is executed by the ITS in the GIC-500 while some of the LPIs on the collection specified are enabled.

Implications

If the erratum occurs, the INVALL command might not disable an enabled LPI, or might not change the priority of an enabled LPI. This means that software might receive an LPI that should have been disabled. It also might not receive an LPI that should have had its priority increased.

Workaround

You can ensure that the erratum does not occur by not relying on the INVALL command to update the settings for enabled LPIs. The functionality of INVALL can be replicated by using a number of INV commands, which are not affected by this erratum. Instead of using the INVALL command, you must instead issue one INV command for each LPI that must use the latest LPI configuration. This will update each LPI individually.

2.6. Category B (Rare)**2.7. Category C****852676: Read of GITS_PIDR3 value may return incorrect value****Category C****Products Affected: GIC-500 Generic Interrupt Ctlr - TERM.****Present in: r1p0****Description**

If you try to modify the Customer Modifiable field of GITS_PIDR3 register then the new value will not be sampled until after another event enables the internal LCB clock

Implications

Software may read 0 from bits [3:0] of GITS_PIDR3 instead of the intended value.

Workaround

Write to any GITS register (other than GITS_TRANSLATER) before reading GITS_PIDR3.

855721: GICD_TYPER.CPUNumber!=0 when the ARE bits are fixed as RAO/WI

Category C

Products Affected: GIC-500 Generic Interrupt Ctlr - TERM.

Present in: r1p1

Description

GICD_TYPER.CPUNumber field reads incorrectly report the number of configured CPUs (topping out at 7) irrespective of whether ARE=0 modes are supported.

Implications

The register value will not be 0 for configurations which don't support ARE=0 unless there is only 1 CPU.

Workaround

Ignore the GICD_TYPE.CPUNumber field as it has no meaning when ARE=1.